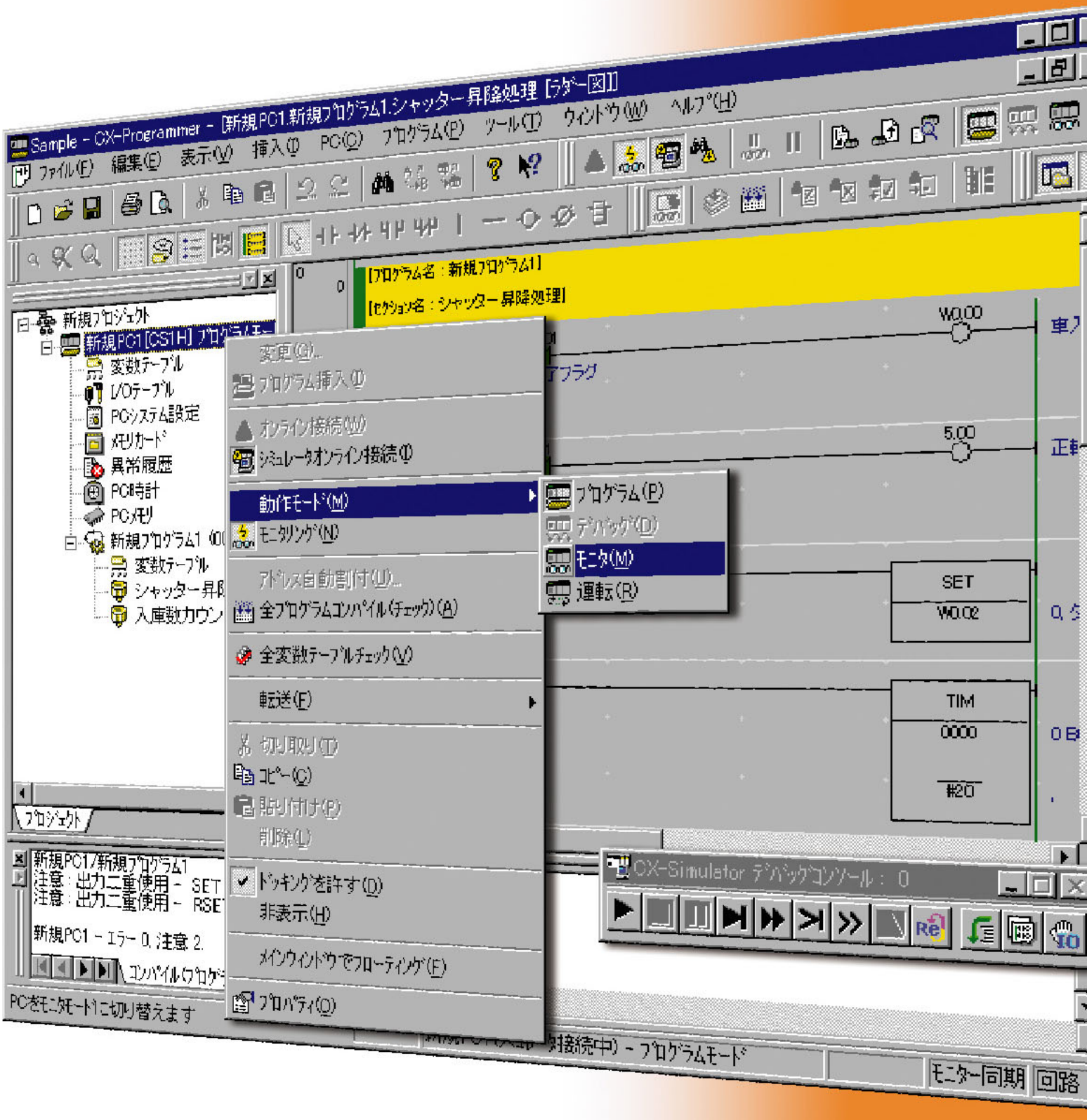
CX-Simulator

導入ガイド



形 WS02-SIMC1-J

- 本誌に記載の無い条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きく影響が予測されるなど、特に安全性が要求される用途への使用をご検討の場合は、定格・性能に対し余裕を持った使い方やフェールセイフ等の安全対策へのご配慮を頂くと共に、弊社営業担当者まで御相談頂き仕様書などによる確認をお願いします。
- 本製品のうち、外国為替及び外国貿易法に定める輸出許可、承認対象貨物（又は技術）に該当するものを輸出（または非居住者に提供）する場合は同法に基づく輸出許可、承認（又は役務取引許可）が必要です。

目　　次

第１章　CX-Simulator の概要

CX-Simulator の概要 ・・・

第1章　CX-Simulatorの概要

CX-Simulator の概要

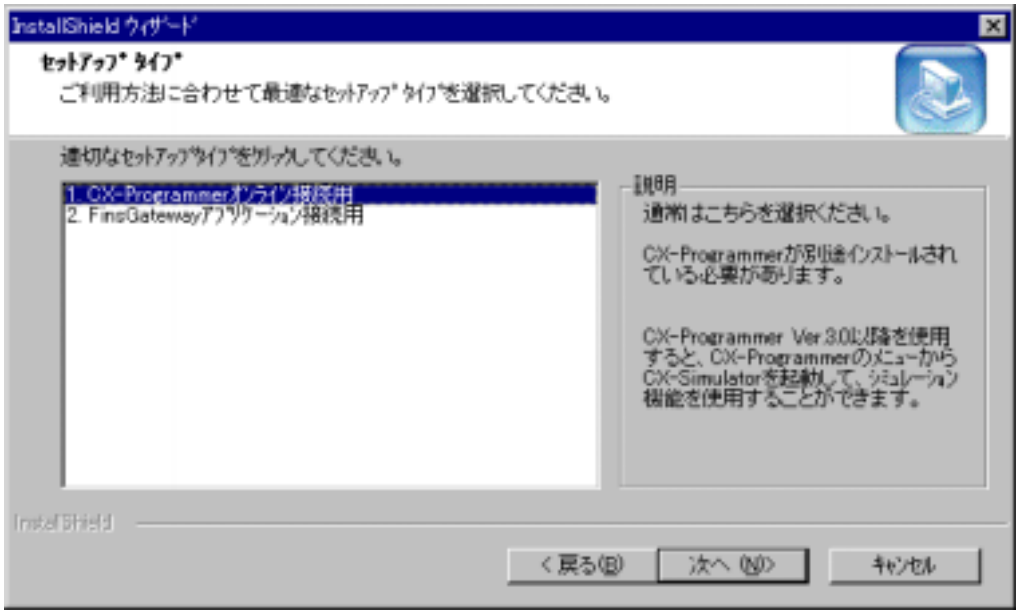
CX-Simulator は、SYSMAC CS/CJ シリーズの CPU ユニットをパソコン内に仮想 PLC として実現し、CPU ユニットと同等に実行（シミュレート）することができます。CX-Programmer と組み合わせて使用することにより、実機 PLC がなくてもパソコン1台でラダープログラムの動作検証やサイクルタイムの事前検証などが行えるようになります。
さらに、CX-Simulator の各種デバッグ機能により、実機 PLC だけでは実現できなかったラダーデバッグが可能となります。

CX-Simulator と CX-Programmer との関係

CX-Simulator はパソコン内の仮想上の PLC ネットワークに仮想 PLC が存在するという状態を作り出します。CX-Programmer Ver.3.0 以降の「シミュレータオンライン接続」機能を使用すると、現在のプロジェクトの PLC 型式で自動的に CX-Simulator の仮想 PLC を起動しオンライン接続することができます。

インストールの前に

CX-Simulator v1.3 では、2 つのセットアップタイプがありますが、CX-Programmer がインストールされているパソコンでは、「1. CX-Programmer オンライン接続用」を選択してください。
CX-Programmer がインストールされているディレクトリに自動的に CX-Simulator がインストールされます。



「2. FinsGateway アプリケーション接続用」は、FinsGateway のネットワークを使用したアプリケーションのデバッグ用として、仮想 PLC を使う場合に選択します。

CX-Simulator の主な特徴

実機 PLC なしでプログラムを実行し、モニタ、デバッグが可能

プログラムの導通モニタ、I/O メモリ現在値モニタが可能。さらに、強制セット／リセット、微分モニタ、データトレース、オンラインエディットなどの通常のデバッグも CX-Programmer から可能です。また、任意のサイクル実行タスクを起動／停止したり、割込タスクを擬似的に起動させることもできます。

実機 PLC なしで、サイクルタイムを確認可能

事前に実機 PLC で実行した場合の計算上のサイクルタイム（現在値、最小値、最大値、平均値、周辺サービス時間 *1）を確認することができます。

プログラムのステップ単位、スキャン単位での実行、I/O ブレーク条件の設定などが可能

実機 PLC では実現できない各種デバッグ機能を用意。

- ステップ実行 ： プログラムを1命令ごとに実行
- スキャン実行 ： プログラムを1スキャン（サイクル）ごとに実行
- I/O ブレーク条件の設定 ： I/O メモリが指定した条件になったときに実行を停止
- スタートポイント、ブレークポイントの指定

表示器やシリアル通信機器と接続したデバッグが可能 *1

パソコンのシリアル通信ポートを PLC の通信ポートとして見たてて、表示器やシリアル通信機器（バーコードリーダ、ID センサなど）と組み合わせたデバッグが可能です。

シリアル通信、ネットワーク通信の送信メッセージを画面表示可能 *1

TXD 命令（通信ポート出力）、SEND 命令（ネットワーク送信）、CMND 命令（コマンド送信）によって発行された送信メッセージを確認することができます。シリアル通信、ネットワーク通信のデバッグに有効です。

*1 [スタート]メニューから CX-Simulator を起動し、仮想 PLC の設定を行なう必要があります。操作の詳細につきましては、第 5 章と、CX-Simulator の取扱説明書を参照してください。

第2章　サンプルプログラムの作成

CX-Programmer を使って、「シャッター開閉による車入庫制御のプログラム」を例題にサンプルプログラムの作成を行います。また、ここで作成したプログラムを使って、第3章以降の CX-Simulator デバッグ機能の使い方を説明します。

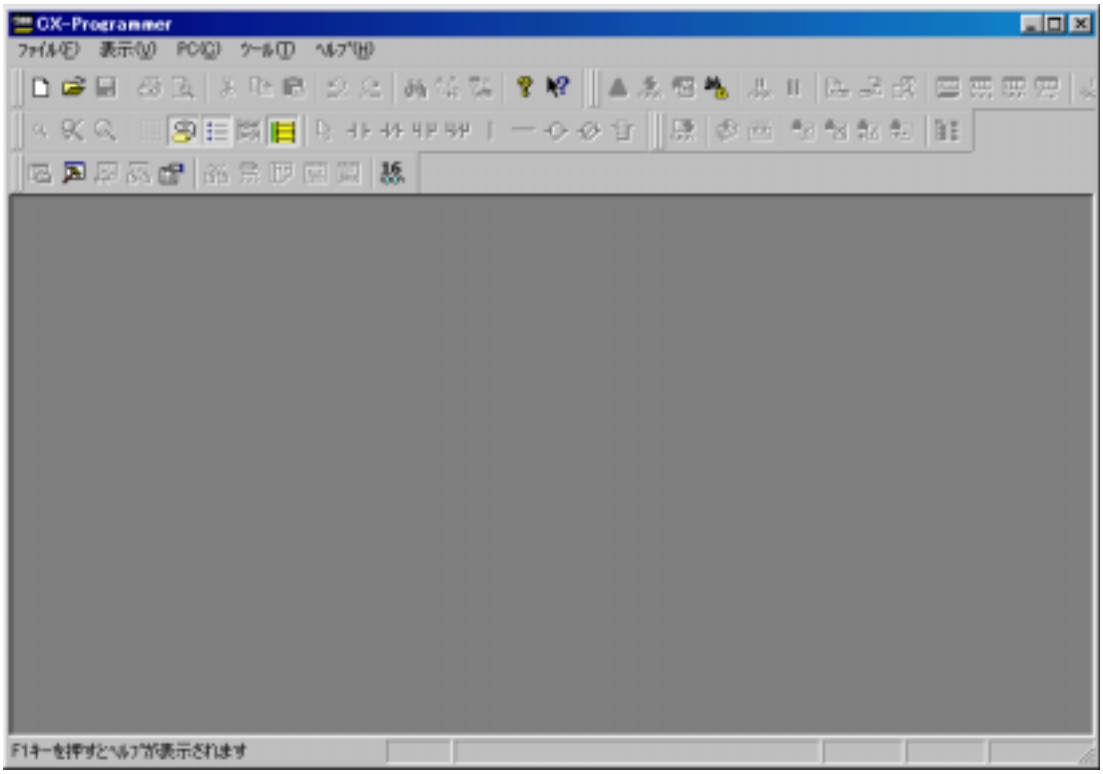
[スタート]
↓
[プログラム]
↓
[Omron]
↓
[CX-Programmer]
↓
[CX-Programmer]

または、CX-Programmer ｱｲｺﾝをﾀﾞﾌﾞﾙｸﾘｯｸします。

2－1　CX-Programmer の起動

Windows より、CX-Programmer を起動します。

CX-Programmer の起動時初期画面が表示されます。

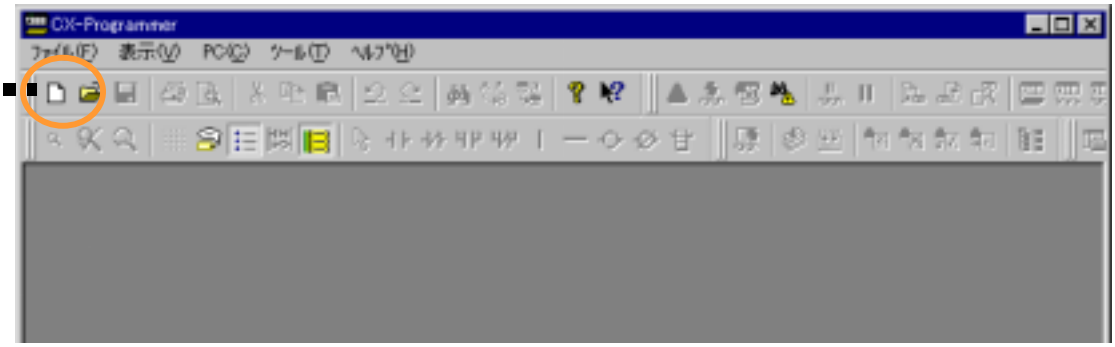


2－2　プロジェクトの新規作成と PC 機種/CPU 形式の設定

CX-Programmer より新規作成ボタンをクリックします。

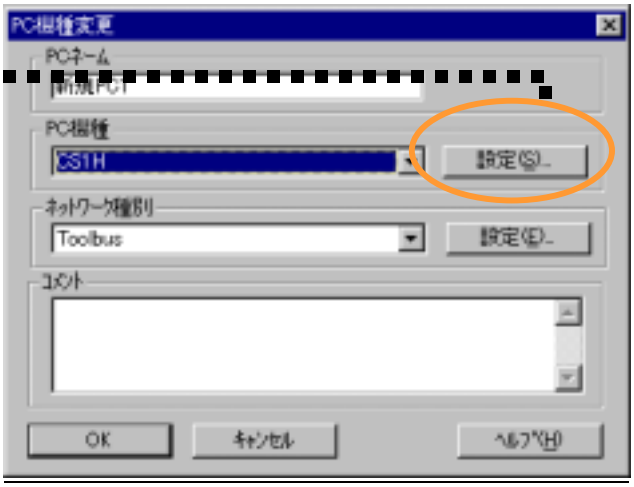
ここでは、PC 機種を“CS1H”、CPU 形式を“CPU63”に設定します。

マウスを左クリック



マウスを左クリック

設定(S)...



設定ボタンの上でマウスを左クリックして、形式設定のメニューを表示させます。

PC 機種の設定をします。
ここでは、“CS1H”に設定します。

▼をクリックしてCPUの形式を選択します。

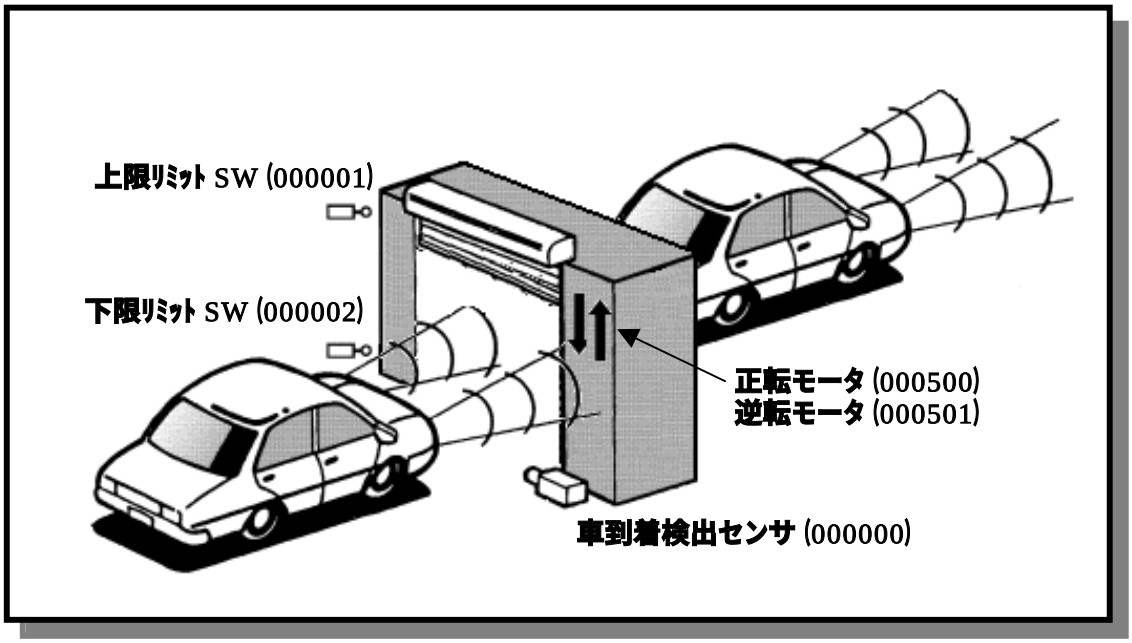
CPU 形式の設定をします。
ここでは、“CPU63”に設定します。

OKをクリックして選択した形式を確定します。

OK

2－3　サンプルプログラムの作成

CX-Programmer を使って簡単なラダープログラムのコーディングを行ないます。
このサンプルプログラムは、シャッター開閉による車入庫制御のプログラムです。
車が接近すると自動的にシャッターを開け、車がゲートを通過すると、2秒後にシャッターを閉じる、というプログラムです。



動作の流れ

①車が入ってくると車到着検出センサ (000000) が ON する。
②シャッターの正転用モータ (000500) が ON する。
③3秒後にシャッターが上限まで達して、上限リミットSW (000001) がONとなり、シャッターの扉が開く。
④車がシャッターを通過すると、車到着検出センサ (000000) が OFF する。
⑤2秒後に逆転用モータ (000501) が ON する。
⑥3秒後にシャッターが下限まで達して、下限リミットSW (000002) がONとなり、シャッターの扉が閉まる。

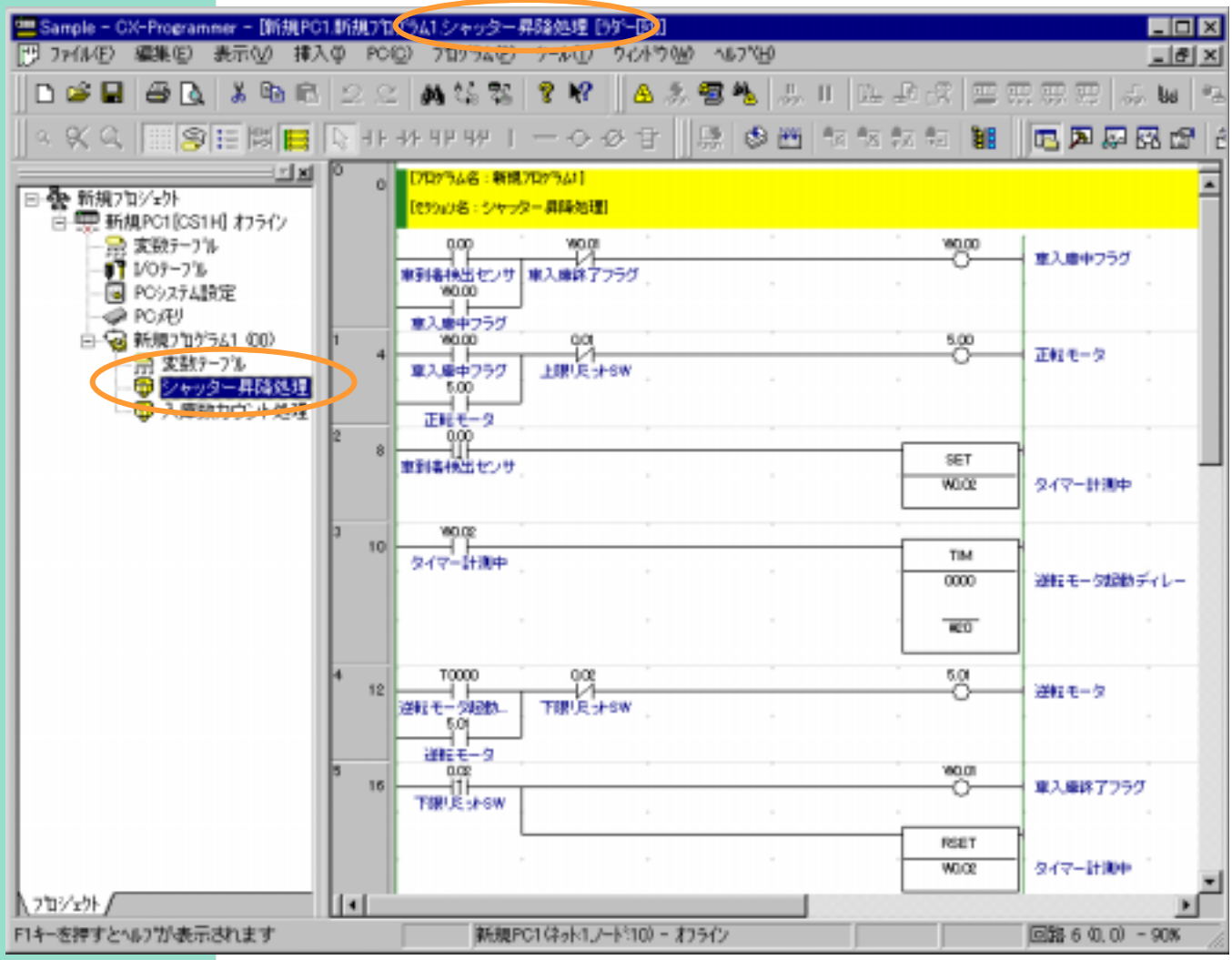
CX-Programmer による具体的なラダープログラムの入力方法については、CX-Programmer 導入ガイドまたは、CX-Programmer の取扱説明書を参照してください。

完成サンプルプログラム

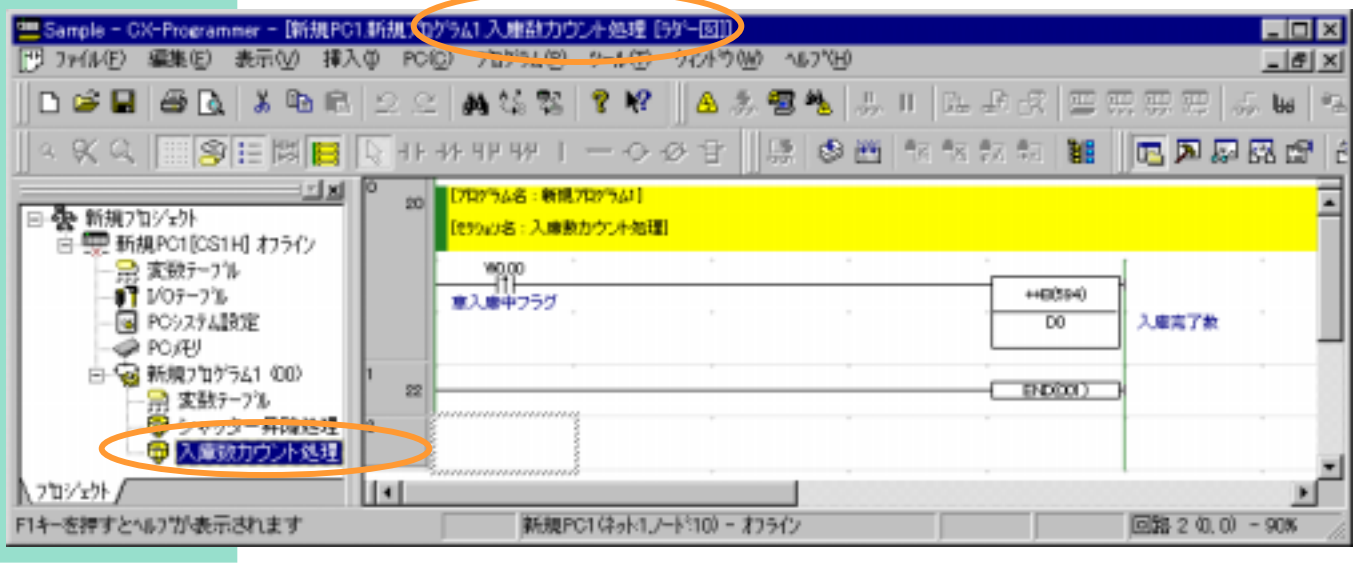
ここでは、CX-Programmer V2 以降のセクション機能により、「シャッター昇降処理」と「入庫数カウント処理」の 2 つにモジュール化します。

本プログラムは処理を簡略化しておりますので、実際のプログラムとは異なります。

シャッター昇降処理



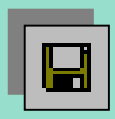
入庫数カウント処理



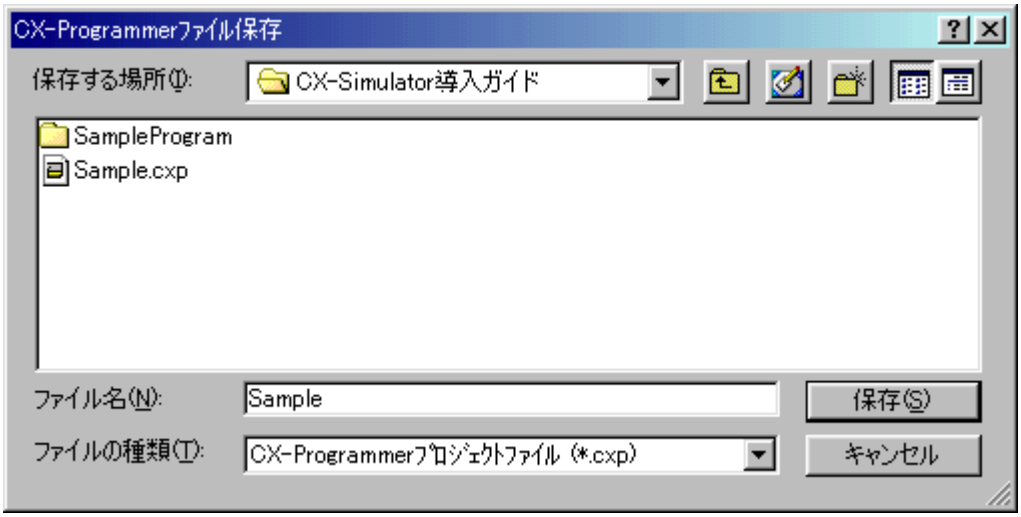
2－4　プログラムの保存

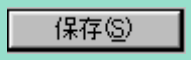
作成したプログラムを保存します。

ここでは、“sample”というファイル名で保存します。



または、[ﾌｧｲﾙ]｜[名前を付けて保存]をメニューから選択します。





プログラムが保存されます。

これ以降の操作で、このサンプルプログラムを使用して CX-Simulator の説明を行います。CX-Programmer はこのまま起動した状態にしておいてください。

第3章
CX-Simulatorによるプログラム実行

CX-Programmer を CX-Simulator 仮想 PLC に接続して、プログラム転送、PLC モード切り替え、サイクルタイム確認、およびラダープログラムの導通モニタなど、CX-Programmer の操作方法を説明します。
なお、CX-Programmer の操作は仮想 PLC に接続している場合でも実機 PLC の操作とまったく同じように使うことができます。操作の詳細につきましては、CX-Programmer の取扱説明書を参照ください。

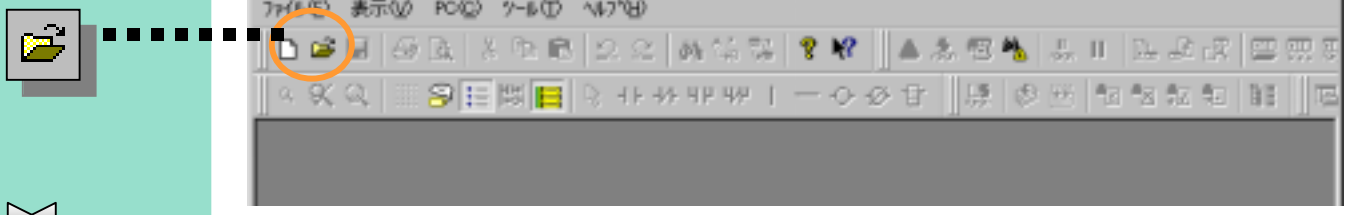
3－1 CX-Simulator 仮想 PLC と CX-Programmer との接続

CX-Simulator の仮想 PLC に CX-Programmer を接続します。

ここでは、第2章により作成したサンプルプログラムを使って説明します。

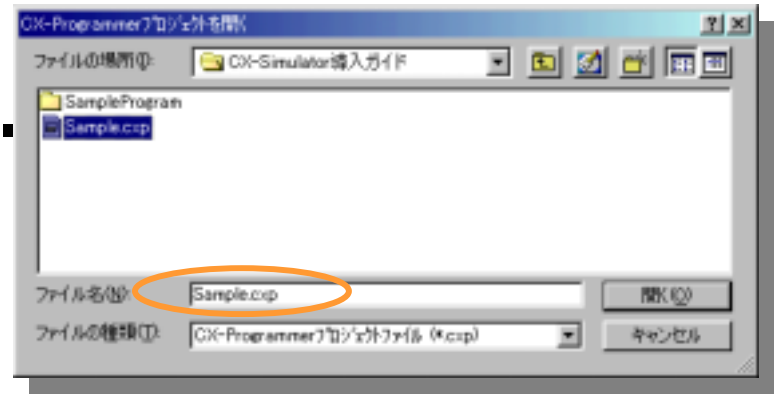
CX-Programmer の起動

CX-Programmer より[ファイルを開く]ボタンをクリックします。

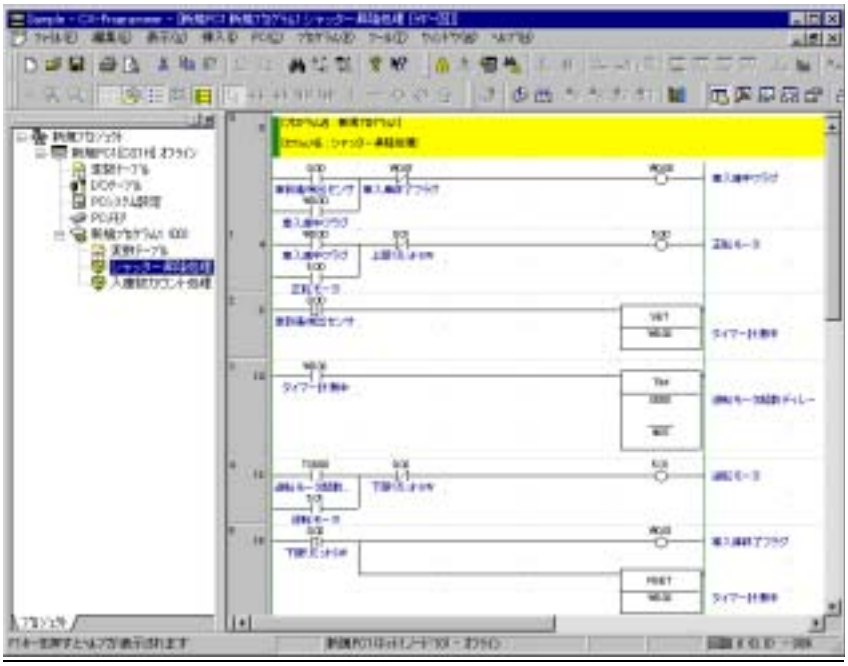


2章で作成したサンプルプログラムを選択します。

“Sample”を選択します。



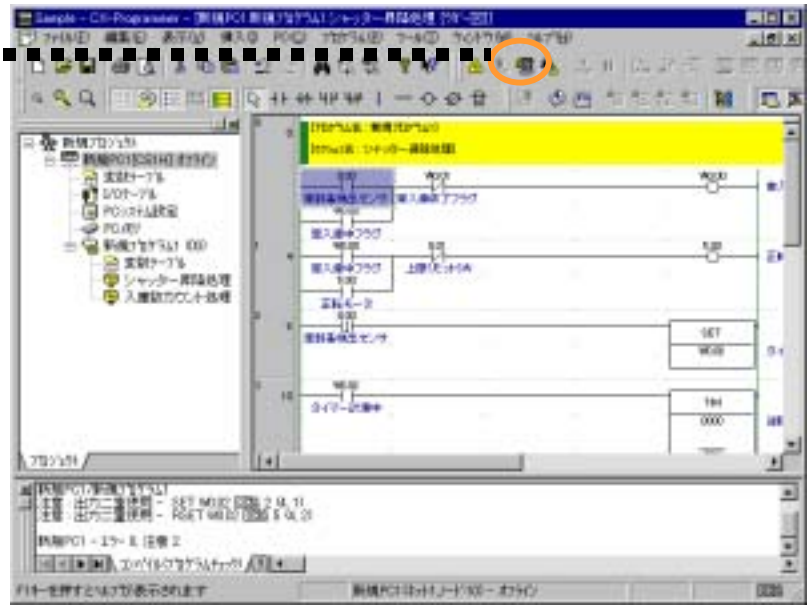
開く(O)



サンプルプログラムがロードされます。

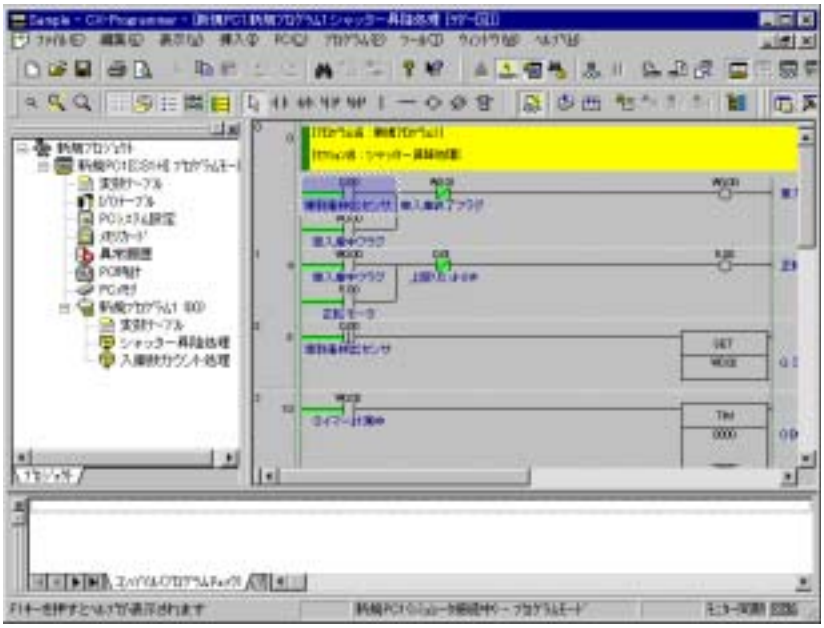
3－2　仮想 PLC へのプログラム転送

シミュレータオンライン接続ボタンをクリックします。



または、[PC]｜[ｼﾐｭﾚｰﾀｵﾝﾗｲﾝ接続]をﾒﾆｭｰから選択します。

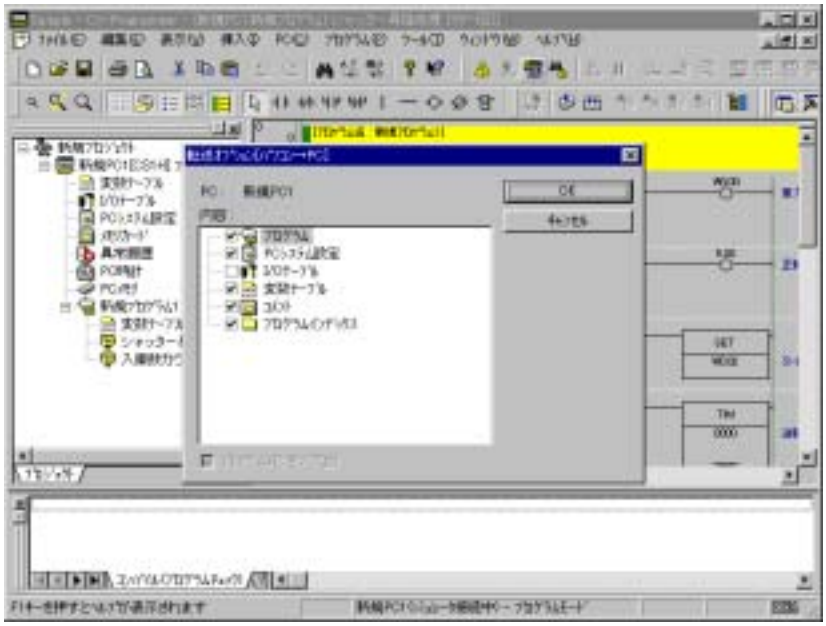
ﾗﾀﾞｰｳｲﾝﾄﾞｳの背景色がｸﾞﾚｰになり、ｵﾝﾗｲﾝ接続されます。



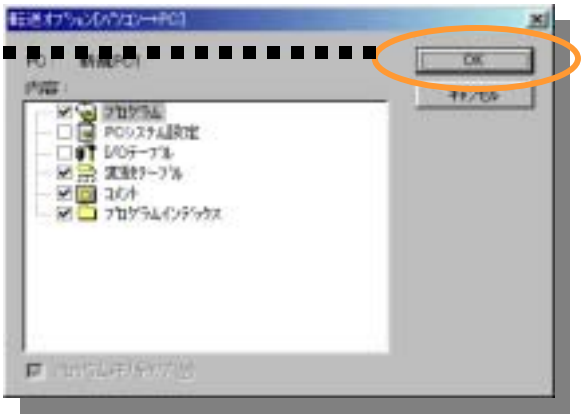
これで、CX-Simulator の仮想 PLC と CX-Programmer が接続しました。

シミュレータオンライン接続により、この後、転送ダイアログが自動的に表示され、ラダープログラムを仮想PLCに転送することができますが、CX-Programmerのオプション設定により接続のみを行う動作に変更できます。

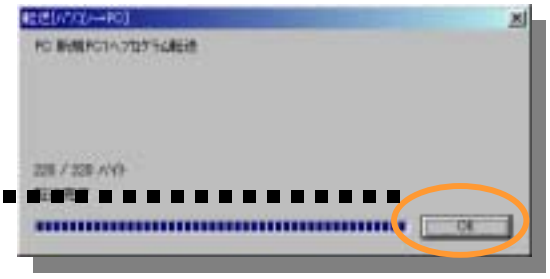
自動的に転送ダイアログが表示されます。



OK



OK



3－3　仮想 PLC の動作モード切り替え

仮想 PLC の動作モードは、CX-Simulator デバッグコンソール画面、あるいは CX-Programmer のどちらからでも行うことができます。

CX-Simulator デバッグコンソールでの操作

プログラムの実行開始（モニタモード）

（運転開始：ﾓﾆﾀﾓｰﾄﾞ）

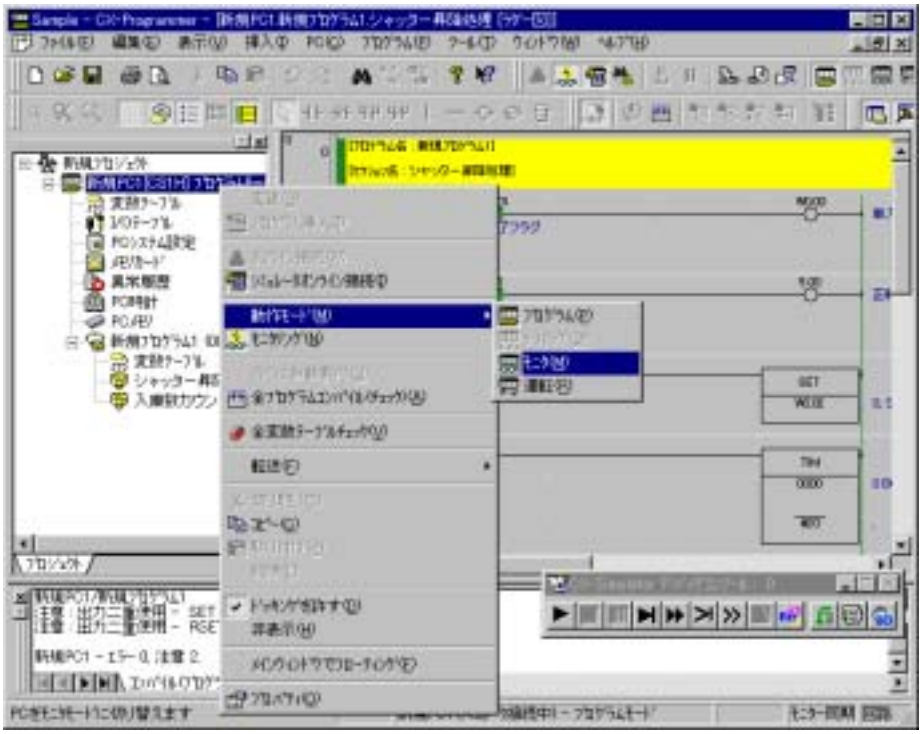
プログラムの実行停止（プログラムモード）

（運転停止：ﾌﾟﾛｸﾞﾗﾑﾓｰﾄﾞ）

デバッグコンソールのタイトルバーに表示されている数字は仮想 PLC での現在のサイクルカウント数になります。

CX-Programmer での操作

新規PLC1ｱｲｺﾝを右クリックし、[動作ﾓｰﾄﾞ]を選択します。
または、メニューから、[PC]｜[動作ﾓｰﾄﾞ]を選択します。



3－4 仮想 PLC でのサイクルタイム確認

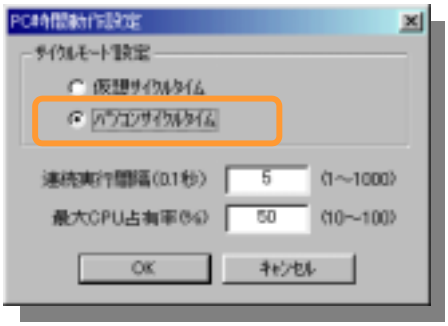
仮想 PLC の動作モードをモニタモードに切り替えます。

CX-Programmerのメニューから、[PC]｜[PC情報]｜[ｻｲｸﾙﾀｲﾑ]を選択します。

仮想サイクルタイムとパソコンサイクルタイムについて

仮想サイクルタイムは、1スキャンごとにトレースしたデータを入力として動作させたいときのようにより高い精度でシミュレートする場合に設定します。このとき、1スキャンの時間軸を拡大して実行しますので、TIM 命令などの現在値の進み具合はパソコンの性能で変わります。

一方、パソコンサイクルタイムは、仮想サイクルタイムよりも精度は低いですが、TIM 命令などの見た目の動作を検証する場合に設定します。

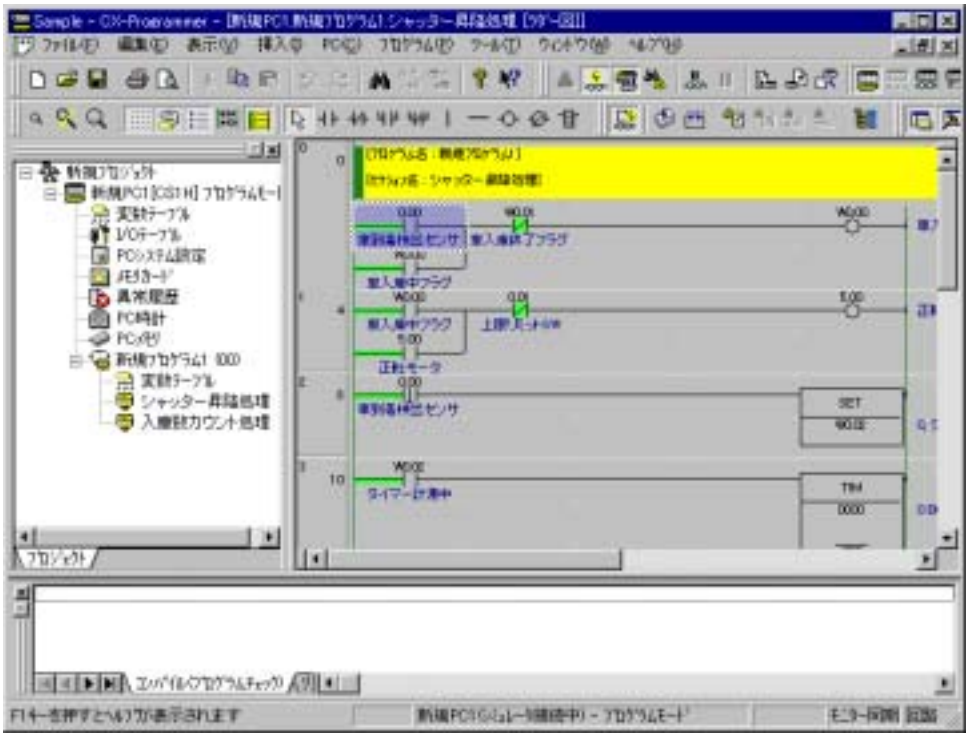


CX-Simulator でのデフォルトのサイクルタイム情報は、「パソコンサイクルタイム」になっています。これは、現在プログラムを実行しているパソコン上でかかった時間です。実機の PLC で動作させたときの計算上のサイクルタイムを確認したい場合には、CX-Simulator の PC 時間動作設定でサイクルモード設定を「仮想サイクルタイム」に変更してください。

以降の操作に先立って、サイクルモード設定を、「パソコンサイクルタイム」に設定しておいてください。

3－5　仮想 PLC でのラダー導通モニタ

CX-Programmer により、ラダーの導通モニタを行います。



ラダー導通モニタの他にも、強制セット／リセット、微分モニタ、データトレース、オンラインエディットなどのデバッグ操作が実機の PLC と接続した場合と同様に CX-Programmer から行なうことが可能です。

第4章
CX-Simulatorによるプログラムデバッグ

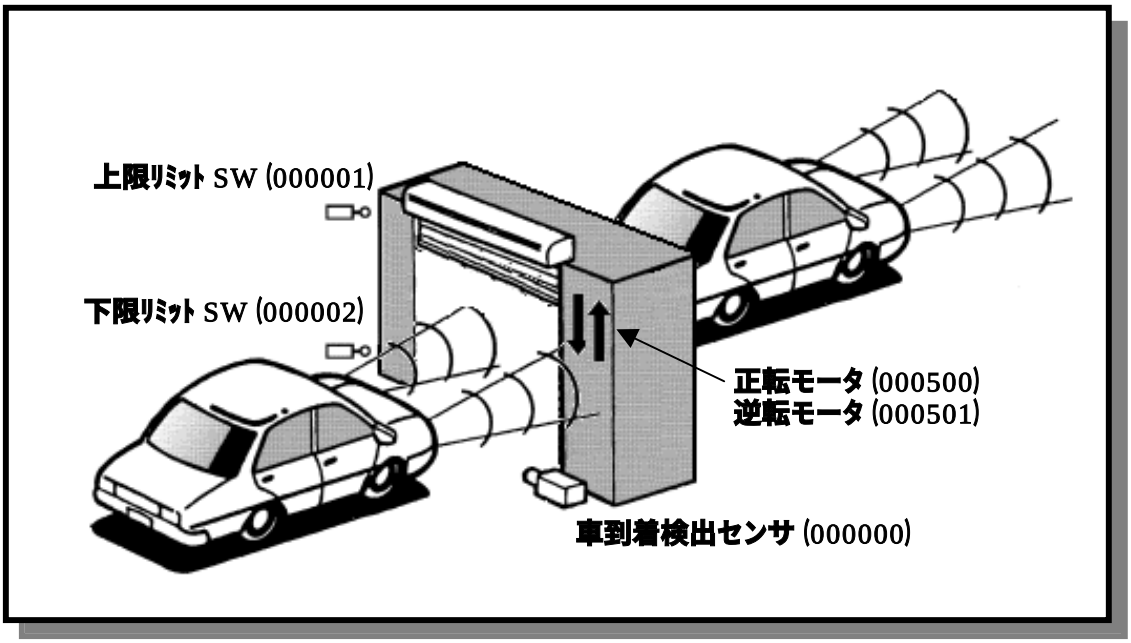
CX-Simulatorの独自の機能を使用して行なう、実機のPLCではできないような、ラダープログラムのデバッグ操作について説明します。

4－1　サンプルプログラムを使ったプログラムデバッグ

CX-Simulator では、実機 PLC のかわりとなる仮想 PLC によるラダープログラム実行ばかりでなく、ワークが移動後のリミットスイッチの ON 動作 **(I/O 条件動作設定)** や、あるメモリの値が指定した値になったときにプログラム実行を停止させる **(I/O ブレーク条件設定)**、任意の範囲のプログラム実行 **(スタートポイント／ブレークポイント設定)** など、さまざまなプログラムデバッグ機能があります。

ここでは、これらの機能を使って、第2章で作成したサンプルプログラム「シャッター開閉による車入庫制御プログラム」を例に、各機能の使い方を説明します。

下図は第2章のサンプルプログラムのイメージ図とその動作の流れです。



動作の流れ

①車が入ってくると車到着検出センサ (000000) が ON する。
②シャッターの正転用モータ (000500) が ON する。
③**3秒後にシャッターが上限まで達して**、上限リミット SW (000001) が ON となり、シャッターの扉が開く。
④車がシャッターを通過すると、車到着検出センサ (000000) が OFF する。
⑤2秒後に逆転用モータ (000501) が ON する。
⑥**3秒後にシャッターが下限まで達して**、下限リミット SW (000002) が ON となり、シャッターの扉が閉まる。

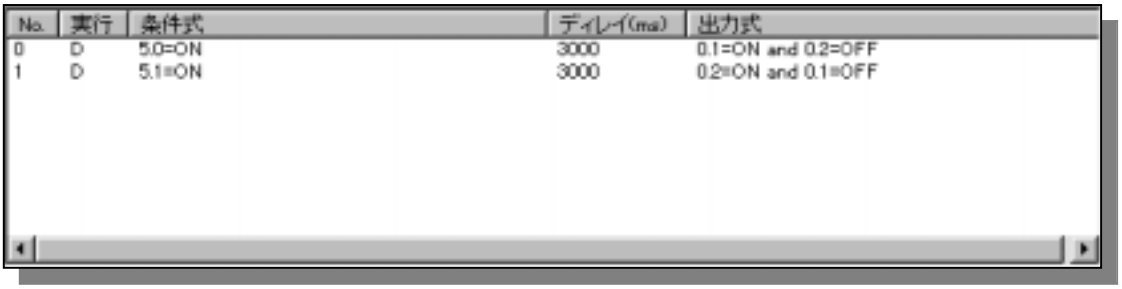
I/O 条件動作設定では、シャッターの上昇、下降動作による上下限リミットスイッチの ON 動作およびそのリミットスイッチが ON するまでの時間を、仮想的な外部入力として設定することができます。この例では、モータが ON した後、3秒後にシャッターのリミットスイッチを ON させる仮想外部入力の指定を使用します。

4－2　仮想外部入力によるデバッグ

CX-Simulator では、例えばシャッターが上昇した後、3秒後に上限リミット SW が ON する、というような外部からの入力（ここでは、上限リミット SW の ON 動作）を設定することができます。
以下に、CX-Simulator の I/O 条件動作ツールの設定方法について説明します。

4－2－1　仮想外部入力の設定内容

ここでは、2つの外部入力信号の動作を設定します。
①正転用モータ（000500）が ON した後、3秒後に上限リミット SW（000001）を ON させる。
（このとき、実際には下限リミット SW（000002）は OFF されています。）
②逆転用モータ（000501）が ON した後、3秒後に下限リミット SW（000002）を ON させる。
（このとき、実際には上限リミット SW（000001）は OFF されています。）



I/O 条件動作ツールでの設定

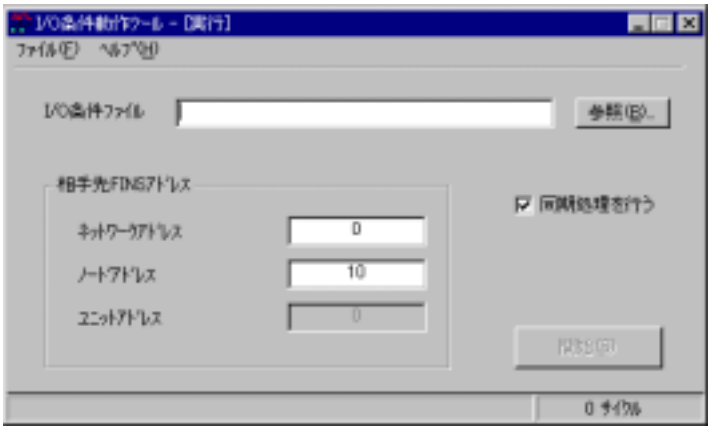
以下に、I/O 条件動作ツールを使って、仮想外部入力の設定について説明します。

4－2－2　I/O 条件動作ツールを起動する

デバッグコンソール画面のポップアップメニューから[IO条件動作]を選択します。

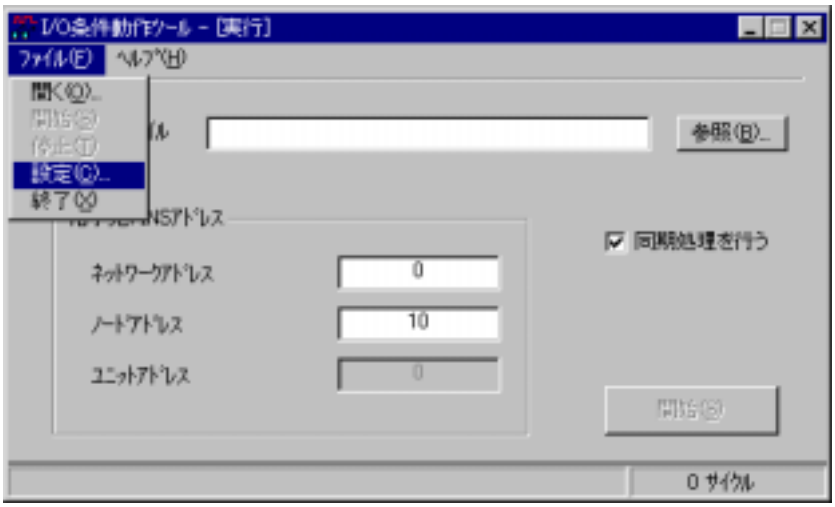
デバッグコンソール画面から、「IO 条件動作」を選択します。

I/O条件動作ツールが起動します。



I/O条件動作ツールの[ファイル]｜[設定]をメニューから選択します。

I/O 条件動作ツールの[ファイル]｜[設定]を選択します。



I/O条件動作[設定]画面が表示します。

4－2－3 I/O 条件動作を設定する

ここでは、下記の I/O 条件動作を設定する例で説明します。

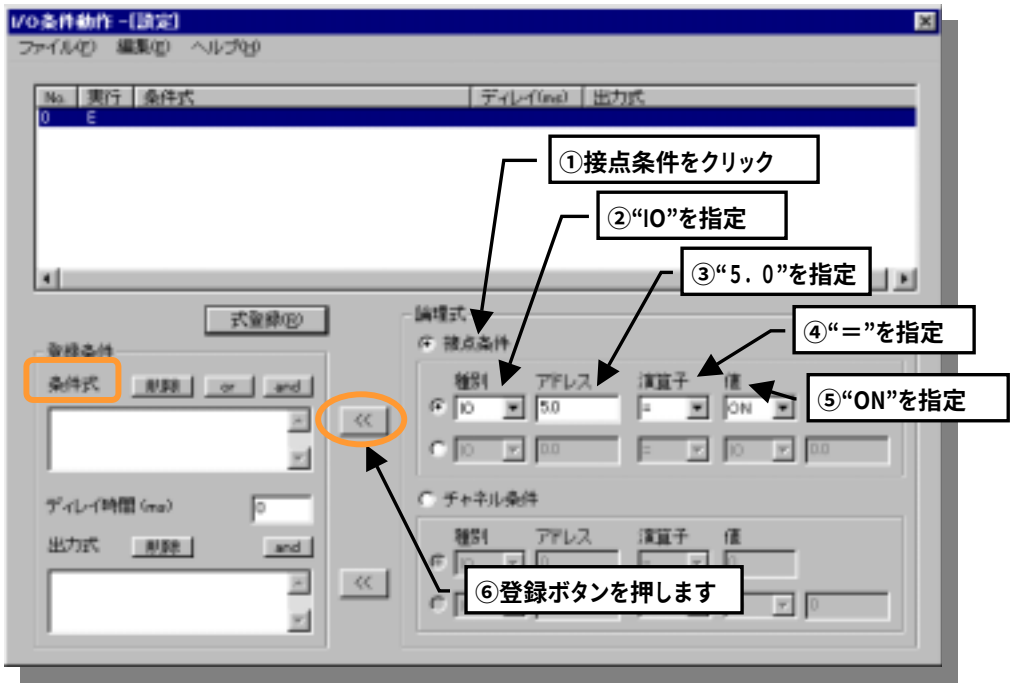
正転用モータ（000500）が ON した後、3秒後に上限リミット SW（000001）を ON させる。
（このとき、実際には同時に下限リミット SW（000002）は OFF されています。）

この例では、条件式、ディレイ時間、出力式は以下のようになります。

- 条件式 ：正転用モータ（000500）が ON する
- ディレイ時間 ：3秒（3000ms）
- 出力式 ：下限リミット SW（000002）は OFF される。 3秒後に上限リミット SW（000001）を ON させる。

条件式の設定

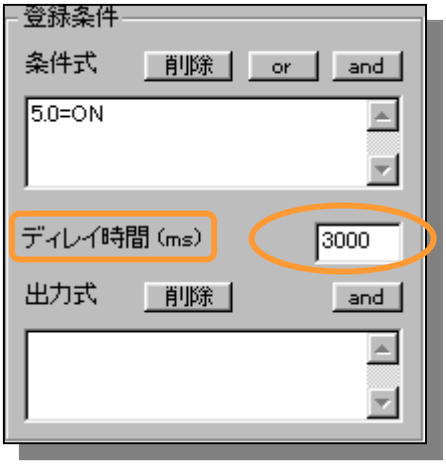
①論理式の“接点条件”をクリックします。
②種別を“IO”にします。
③アドレスを“5.0”にします。
④演算子を“＝”にします。
⑤値を“ON”にします。
⑥登録ボタン << を押します。



ディレイ時間の設定

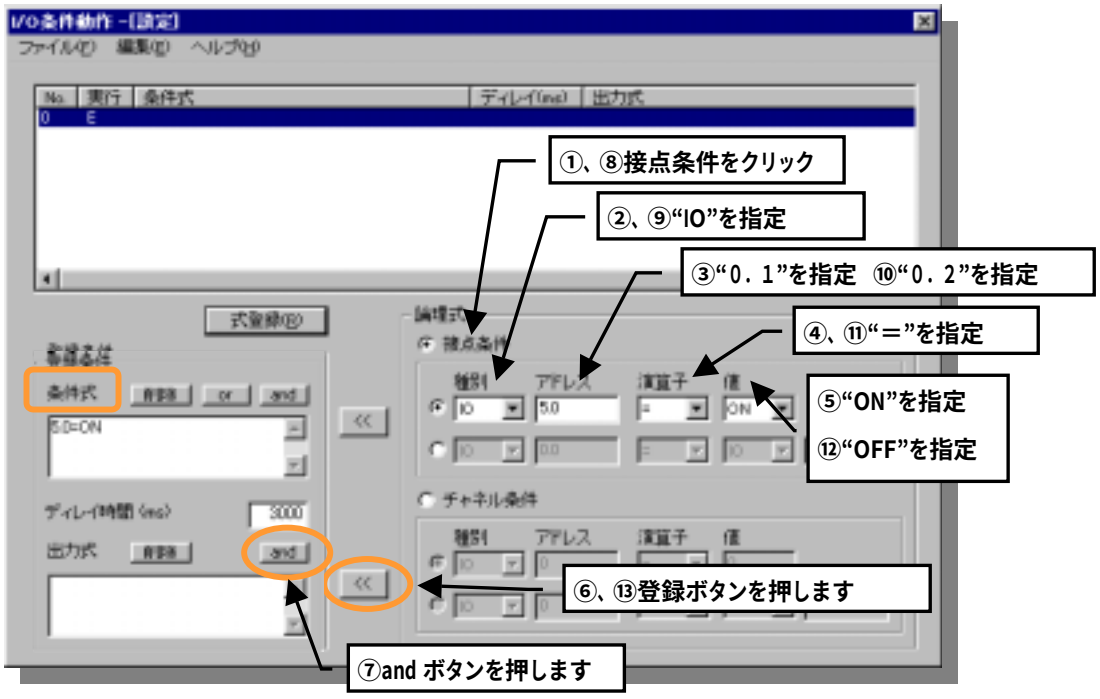
ディレイ時間入力欄に、3000（3秒）を入力します。

ディレイ時間（ms）の欄に、3000ms（＝3秒）と設定します。

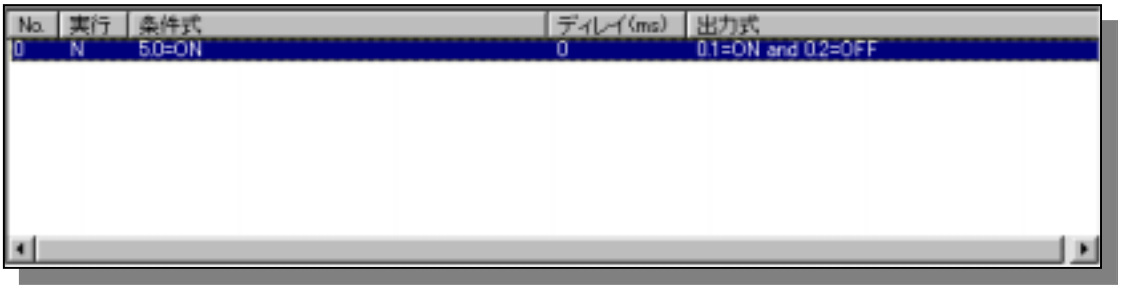


出力式の設定

①論理式の“接点条件”をクリックします。
②種別を“IO”にします。
③アドレスを“0.1”にします。
④演算子を“＝”にします。
⑤値を“ON”にします。
⑥登録ボタン << を押します。
⑦ and ボタンを押します。
⑧論理式の“接点条件”をクリックします。
⑨種別を“IO”にします。
⑩アドレスを“0.2”にします。
⑪演算子を“＝”にします。
⑫値を“OFF”にします。
⑬登録ボタン << を押します。



式登録(R) ■■■■■■ [式登録]ボタンを押します。

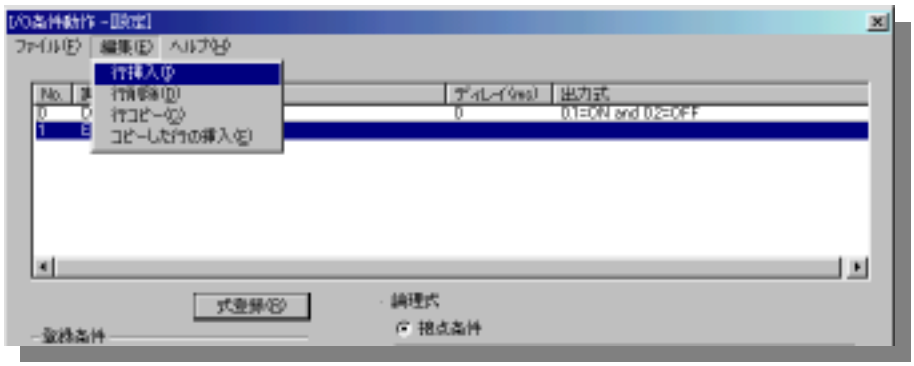


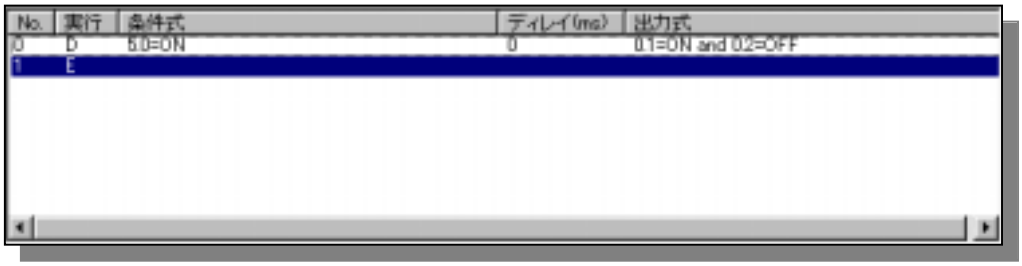
登録された式をダブルクリックします。■■■■■■■■

実行欄が、“N”から“D”に変わります。これにより、この式が実行されることになります。

I/O条件動作[設定]の[編集]｜[行挿入]をメニューから選択します。

2つ目の式を登録するために、新たな行を追加します。





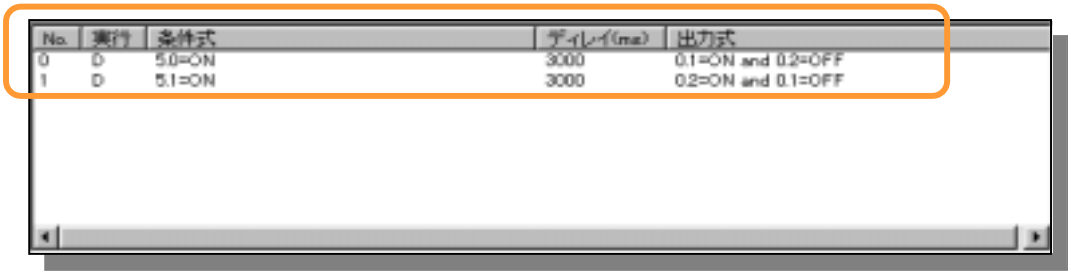
1つ目の式登録の操作と同様に下記の設定も行ないます。

逆転用モータ(000501)が ON した後、3秒後に下限リミット SW(000002)を ON させる。
(このとき、実際には同時に上限リミット SW(000001)は OFF されています。)

この例では、条件式、ディレイ時間、出力式は以下のようになります。

- 条件式　:逆転用(000501)モータが ON する。
- ディレイ時間　:3秒(3000ms)
- 出力式　:上限リミット SW(000001)は OFF される。　3秒後に下限リミット SW(000002)を ON させる。

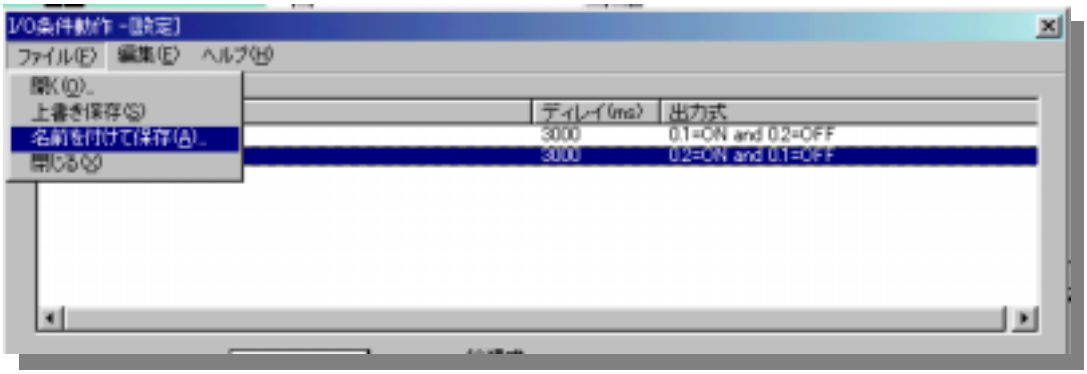
これら2つの式を登録完了した後の設定画面は下記のようになります。



条件式、出力式は、各設定欄に直接キーボードから式を入力することもできます。

4－2－4 I/O 条件動作内容を保存して終了する

I/O条件動作[設定]の[ﾌｧｲﾙ]｜[名前を付けて保存]をﾒﾆｭｰから選択します。



ﾌｧｲﾙ名に“sample”と入力します。

保存(S)

I/O条件動作[設定]の[ﾌｧｲﾙ]｜[終了]をﾒﾆｭｰから選択します。

I/O 条件動作ツール[設定]を終了します。

4－2－5　仮想外部入力によるプログラムデバッグ

I/O条件動作による外部入力デバッグの準備

あらかじめ下記に示す状態にします。

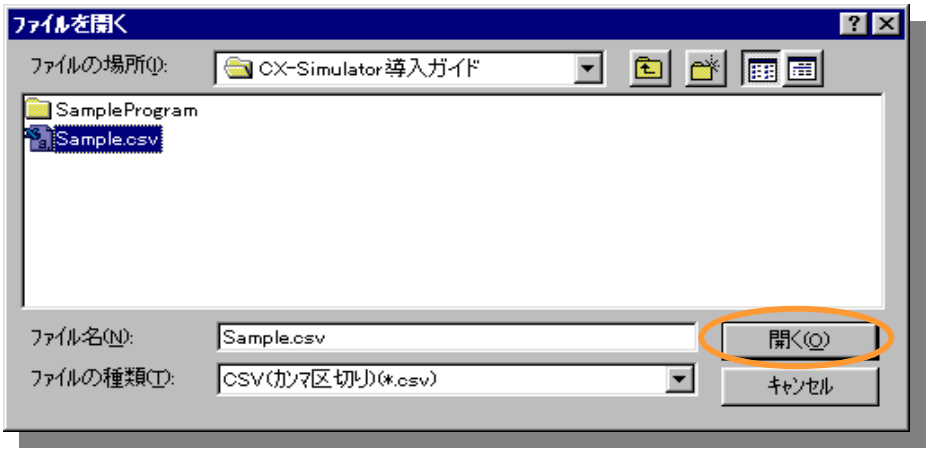
- 第2章で作成したサンプルプログラム (Sample.cxp) を CX-Programmer にロードします。
- 第3章の手順により、CX-Programmer と CX-Simulator 仮想 PLC をオンライン接続し、仮想 PLC にサンプルプログラムを転送します。
- 仮想 PLC のラダー実行を開始します。

これで、準備は完了です。

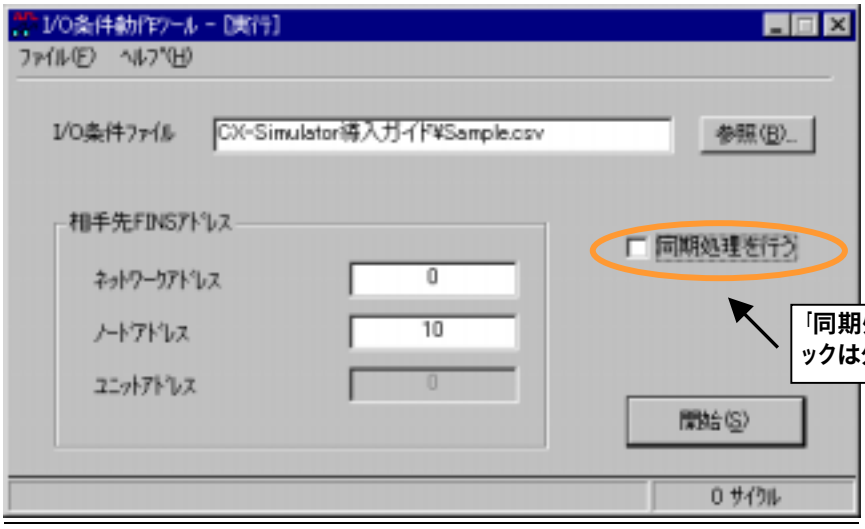
I/O 条件動作ツールの実行

I/O条件動作[実行]の[ファイル]｜[開く] メニューから“Sample.csv”を選択します。

I/O 条件動作ツール[設定]で設定したファイル (Sample.csv) を読み込みます。



「同期処理を行う」のチェックを外します。



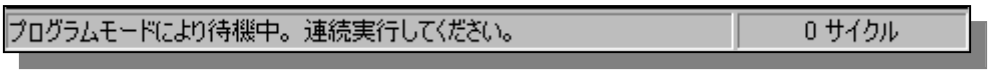
「同期処理を行なう」のチェックは外します。

開始(S)

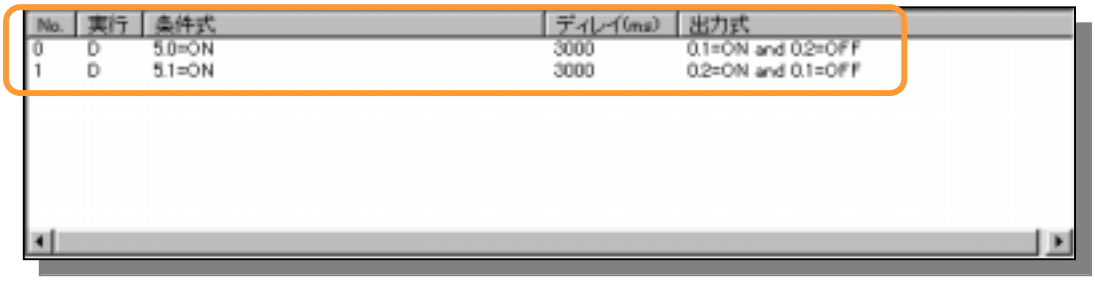
開始ボタンを押すと、設定した条件が成立かどうかの I/O 条件監視が開始されます。

I/O 条件の監視が開始されると、「I/O 条件監視実行中」と表示され、サイクルが加算されていきます。

仮想 PLC のラダー実行が開始されていない場合は以下のように表示されます。このときは、仮想 PLC のラダー実行を開始してください。

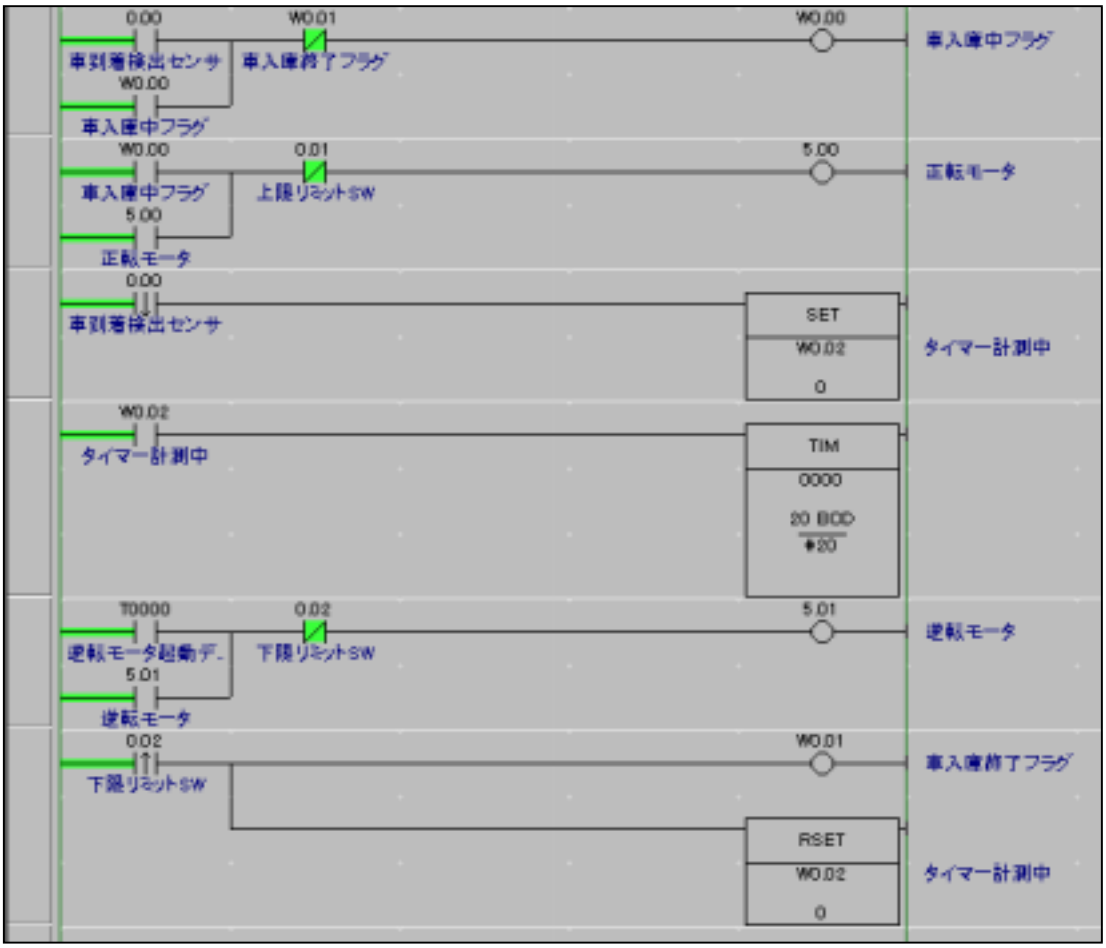


上記により、下記の 2 つの条件式の成立および、成立後の出力動作を開始しています。



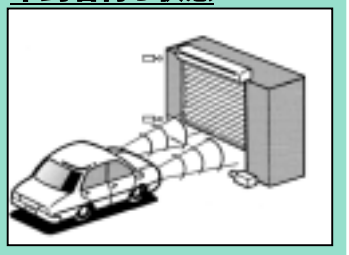
CX-Programmer のラダーウインドウを表示し、「シャッター昇降処理」を表示し、導通モニタを行います。

または、CX-Programmerの[PC]｜[ﾓﾆﾀ]　｜[ﾓﾆﾀﾘﾝｸﾞ]をﾒﾆｭｰから選択します。

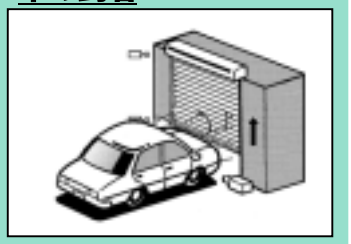


これまでの操作により、仮想 PLC によりラダーが実行されており、I/O 条件動作で設定した条件成立を監視している状態になっています。
これ以降は、実際に車の到着、ゲート通過を擬似入力し、ラダープログラムのデバッグを行っていきます。

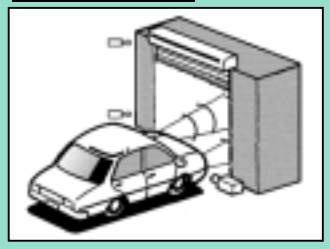
車到着待ち状態



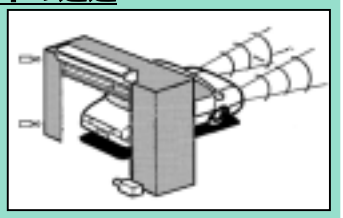
車の到着



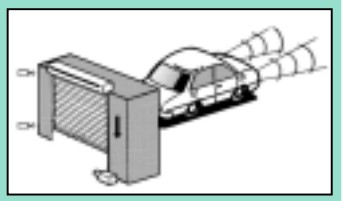
シャッター上昇



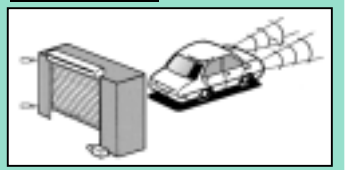
車の通過



シャッター下降



車入庫完了



4－2－6　仮想外部入力を使ったプログラムデバッグ操作例

以下の処理の流れを順に確認していきます。

①車到着待ち状態

②車の到着 — CX-Programmerにより、車到着検出ｾﾝｻ (000000) をセットします。

③シャッター上昇開始

④3秒経過後 — I/O条件動作で設定した条件式および出力式により、上昇ﾓｰﾀ (000500) がONし、3秒後 (ｼｬｯﾀｰ上昇完了後) 、上限ﾘﾐｯﾄSWをONさせます。

⑤シャッター上昇完了

⑥車の通過 — CX-Programmerにより、車到着検出ｾﾝｻ (000000) をリセットします。

⑦2秒経過後 — ラダープログラムにより、車通過後、2秒をカウントします。

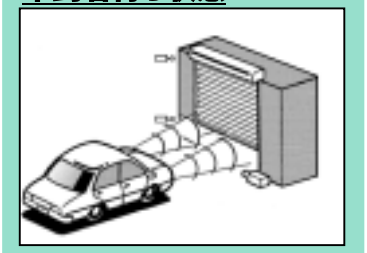
⑧シャッター下降開始

⑨3秒経過後 — I/O条件動作で設定した条件式および出力式により、下降ﾓｰﾀ (000501) がONし、3秒後 (ｼｬｯﾀｰ下降完了後) 、下限ﾘﾐｯﾄSWをONさせます。

⑩シャッター下降完了

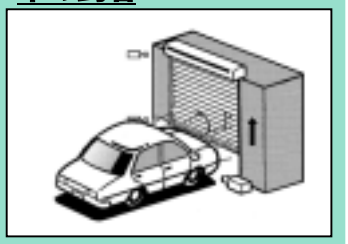
⑪車入庫完了 — ラダープログラムにより、車入庫数をカウントします。

車到着待ち状態



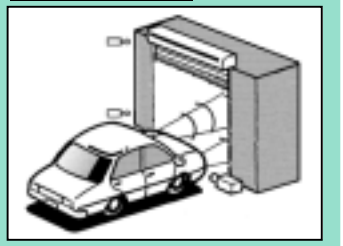
接点0.00をクリック後、右クリックし、[ｾｯﾄ/ﾘｾｯﾄ] | [ｾｯﾄ] を選択します。

車の到着



車到着検出センサ(0.00)をONさせることにより、正転モータ(5.00)がONします。

シャッター上昇

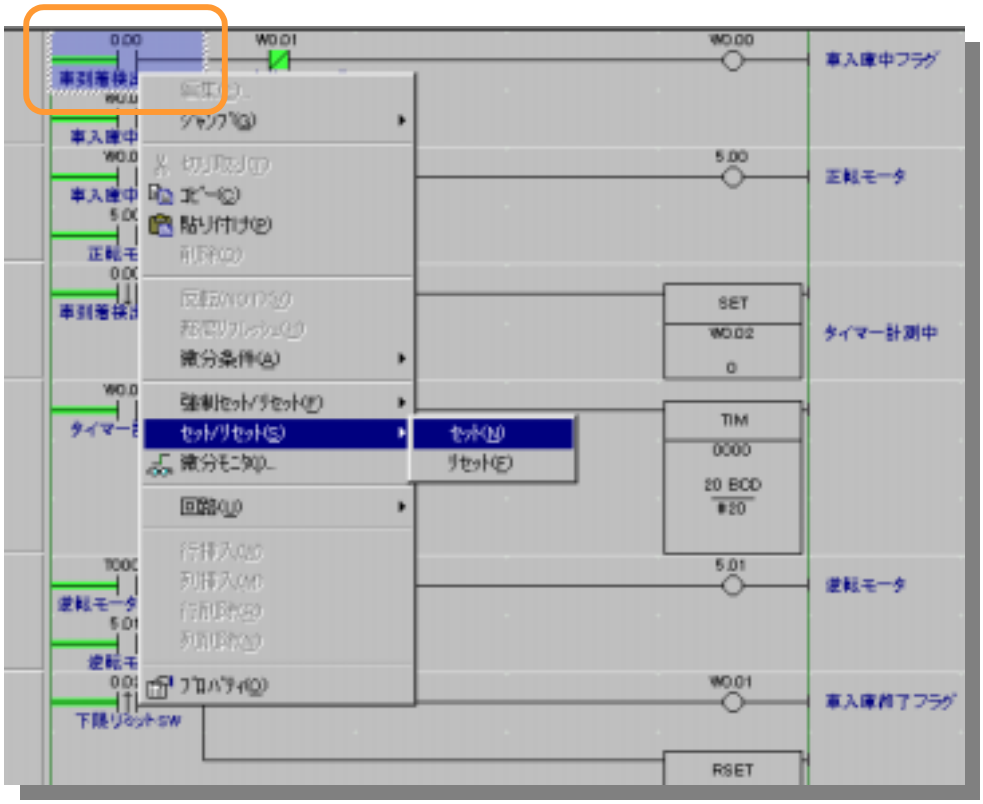


車到着の擬似入力による動作確認

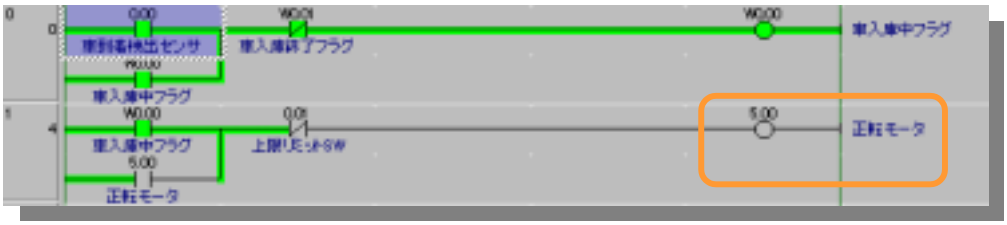
これ以降は、処理の流れに従ってプログラムデバッグを始めていきます。

まず、車の到着待ち状態から車到着の状態を、CX-Programmer のセット機能により行っていきます。

車到着の状態を、車到着検出センサ(000000)を ON することにより行います。



車到着の状態(車到着検出センサをセット)すると、正転モータが ON します。

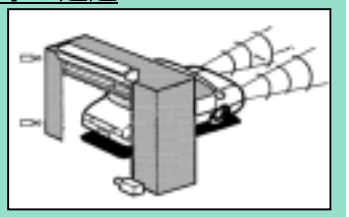


動作しない場合は、仮想 PLC の動作モードを一旦プログラムモードに切り替えた後、再度同じ操作を行ってください。

3秒経過後、シャッターが上限リミット SW を ON させ、正転モータが OFF します。

(I/O 条件動作で設定した条件式と出力式により、上限リミット SW を ON させます。)

車の通過

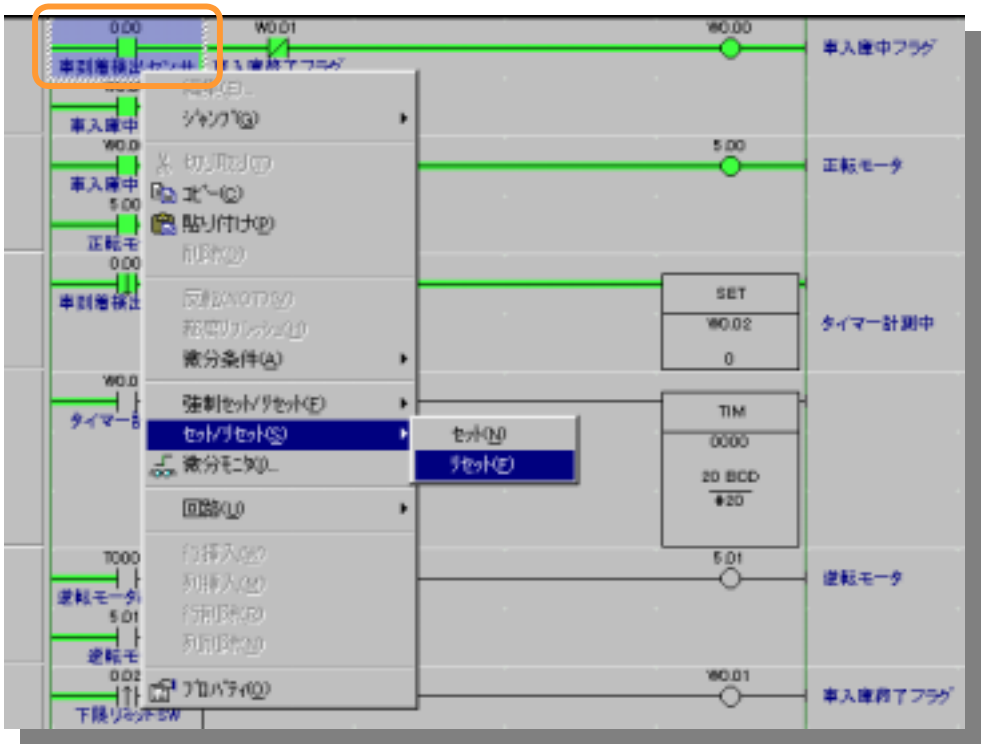


接点0.00をクリック後、右クリックし、[セット/リセット] | [リセット] を選択します。

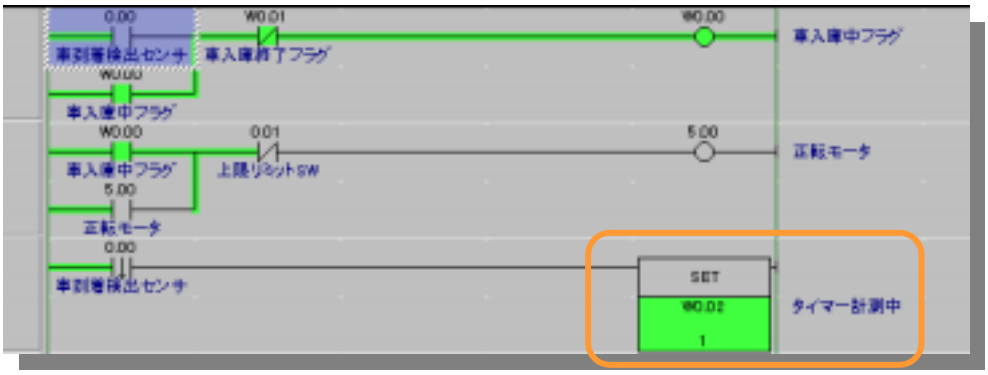
車通過後の擬似入力による動作確認

次は、車がシャッターを通過する状態を、CX-Programmer のリセット機能により行っていきます。

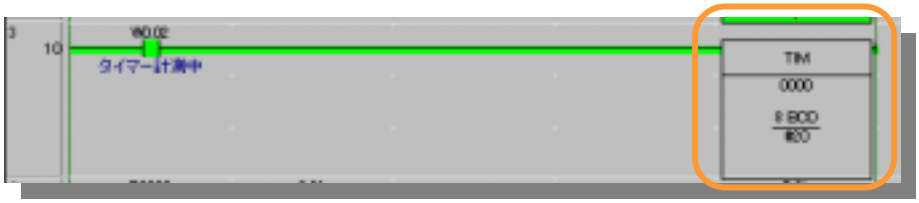
車通過の状態を、車到着検出センサ (0.00) を OFF することにより行います。



車通過の状態 (車到着検出センサをリセット) すると、タイマー計測中フラグが ON します。



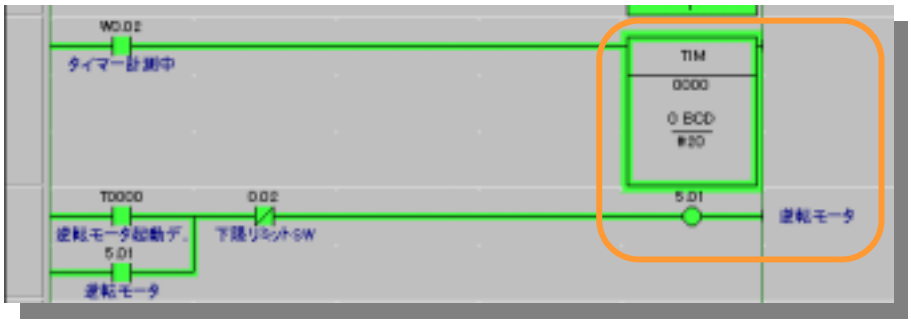
車到着検出センサ (0.00) をOFFさせることにより、タイマー計測中フラグ (W0.02) がONします。



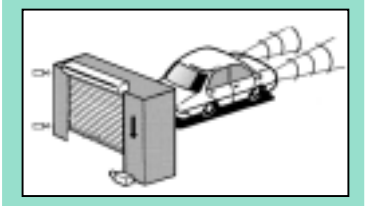
タイマーが計測を開始します。

タイマーが2秒の計測を完了後、逆転モータ(5.01)がONします。

タイマーが2秒計測すると、タイマーフラグが ON し、逆転モータが ON します。

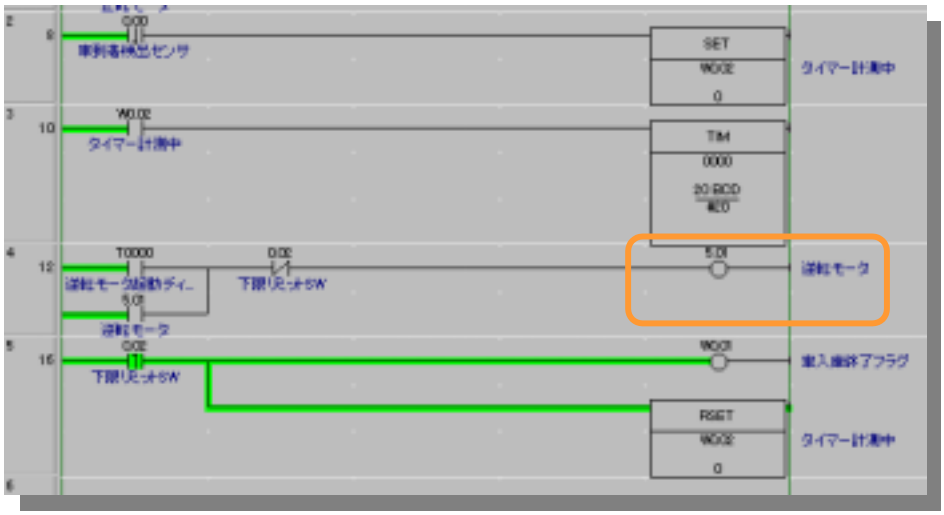


シャッター下降



3秒経過後、シャッターが下限リミット SW を ON させ、逆転モータが OFF します。

(I/O 条件動作で設定した条件式と出力式により、下限リミット SW を ON させます。)



以上の操作により、車到着から車通過までの一連の動作をデバッグすることができます。

このように、車の到着、通過の状態を CX-Programmer のセット／リセットの機能により行うことができます。
また、I/O 条件動作により、シャッターの上昇／下降動作を自動的に作り出すことができます。

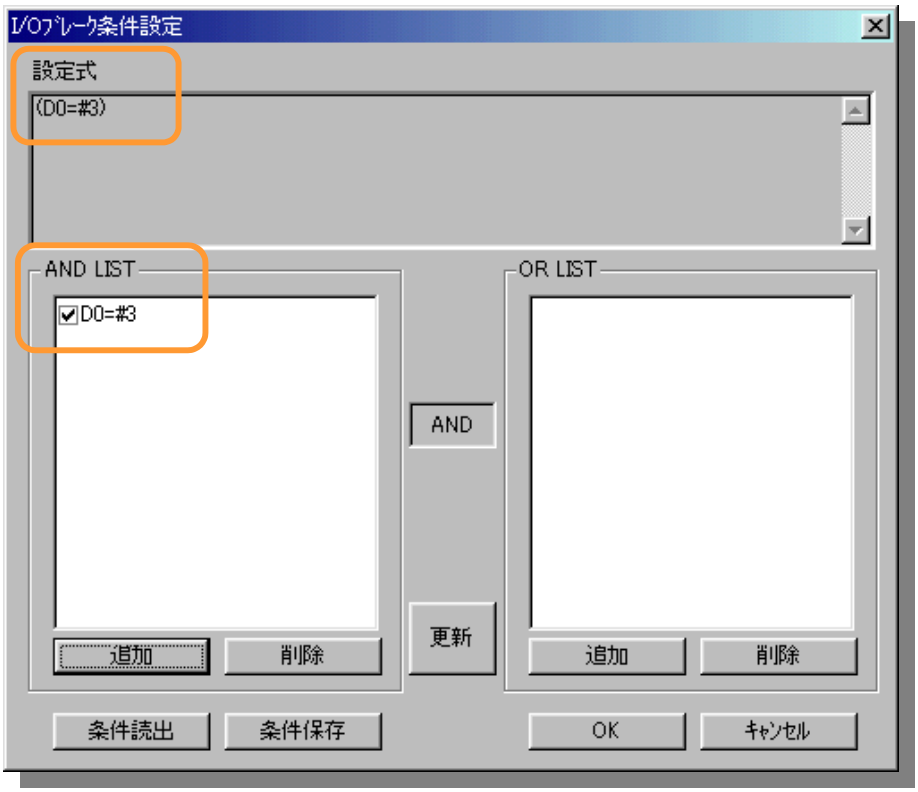
4－3 I/O ブレーク条件設定によるデバッグ

CX-Simulator では、例えば入庫した車の台数がある値になったときにプログラムを停止させることができます。これは、I/O メモリの状態を監視し、設定した条件が成立したときにブレークする機能です。

4－3－1 I/O ブレーク条件の設定内容

ここでは、I/O ブレーク条件設定を使って、車の入庫数が3台になったときにプログラムを停止させる方法について説明します。

車の入庫完了数の格納メモリを D0 (データメモリの0チャネル) とします。

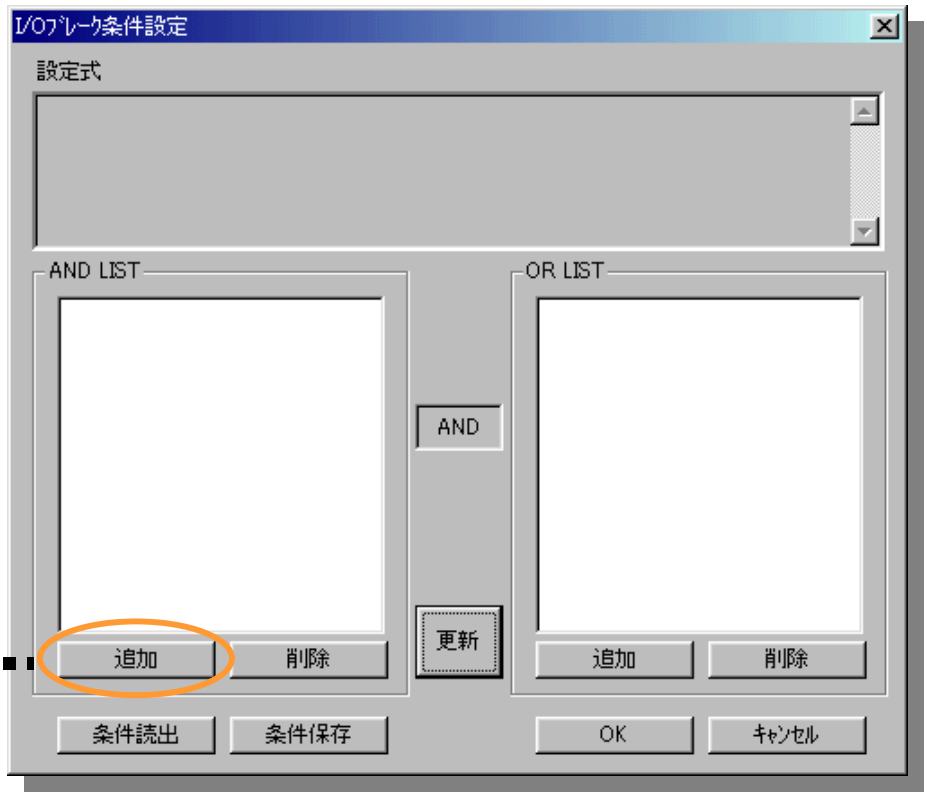


I/O ブレーク条件設定での設定

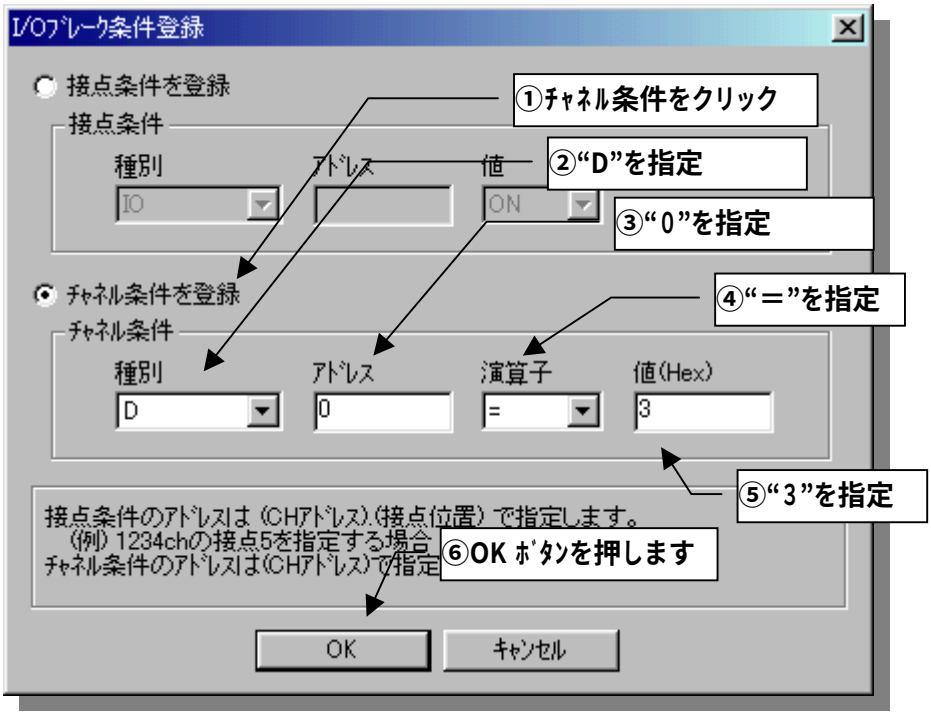
4－3－2 I/O ブレーク条件を設定する

デバッグコンソールから I/O ブレーク条件設定を起動します。

追加



①チャネル条件を登録をクリックします。
②種別を“D”にします。
③アドレスを“0”にします。
④演算子を“＝”にします。
⑤値を“3”にします。
⑥ OK を押します。



リストに登録された式をチェックします。(式を有効にします。)

OK

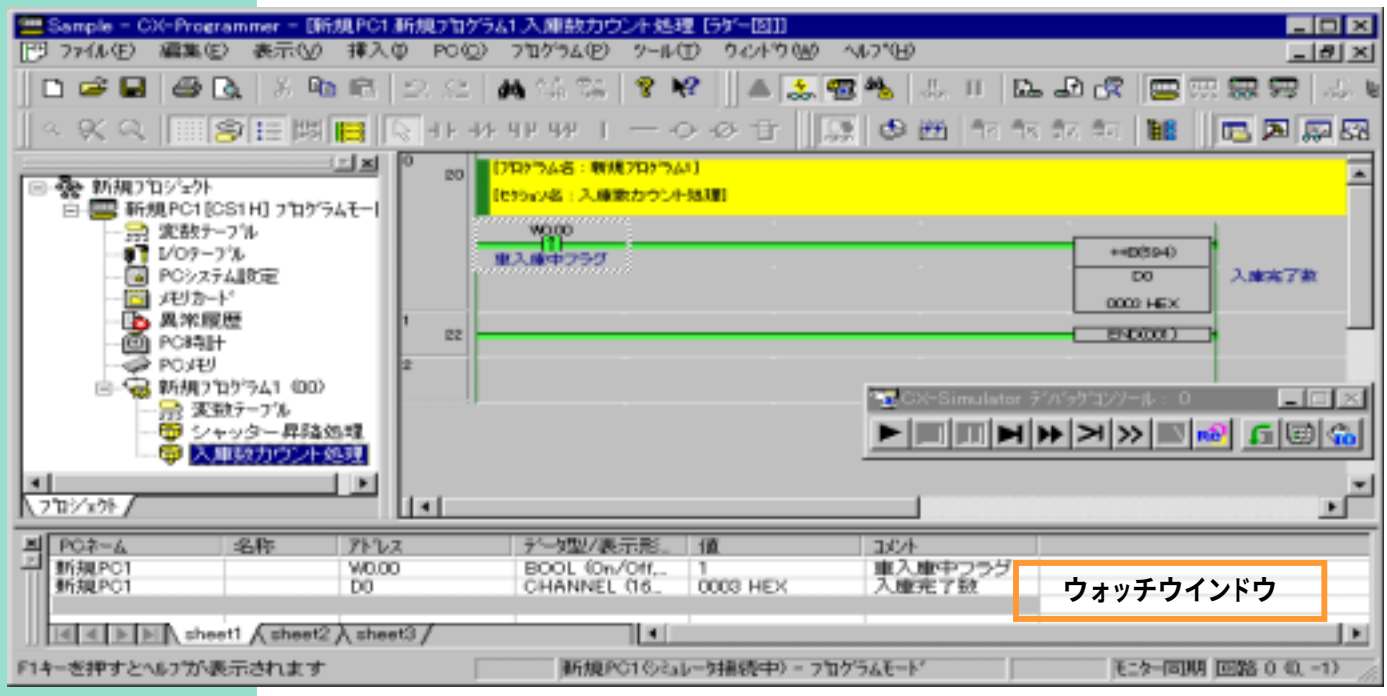
リスト (AND LIST) に登録されます。

これで、車入庫完了数 (D0 の値) が3台 (Hex:#0003) になったときに、プログラムの実行が停止します。

I/O ブレーク条件設定により、あるアドレスの値が指定した値になると、プログラムの実行を一時停止させることができます。
これにより、条件が成立したときの I/O メモリの状態を確認することができます。
CX-Programmer のウォッチウインドウや PC メモリ機能と併用すると一時停止した状態で、すべての I/O メモリの状態を確認することができます。

4－3－3 I/O ブレーク条件を使ったプログラムデバッグ操作例

あらかじめサンプルプログラムの「入庫数カウント処理」をダブルクリックしてラダーモニタ表示にしておきます。



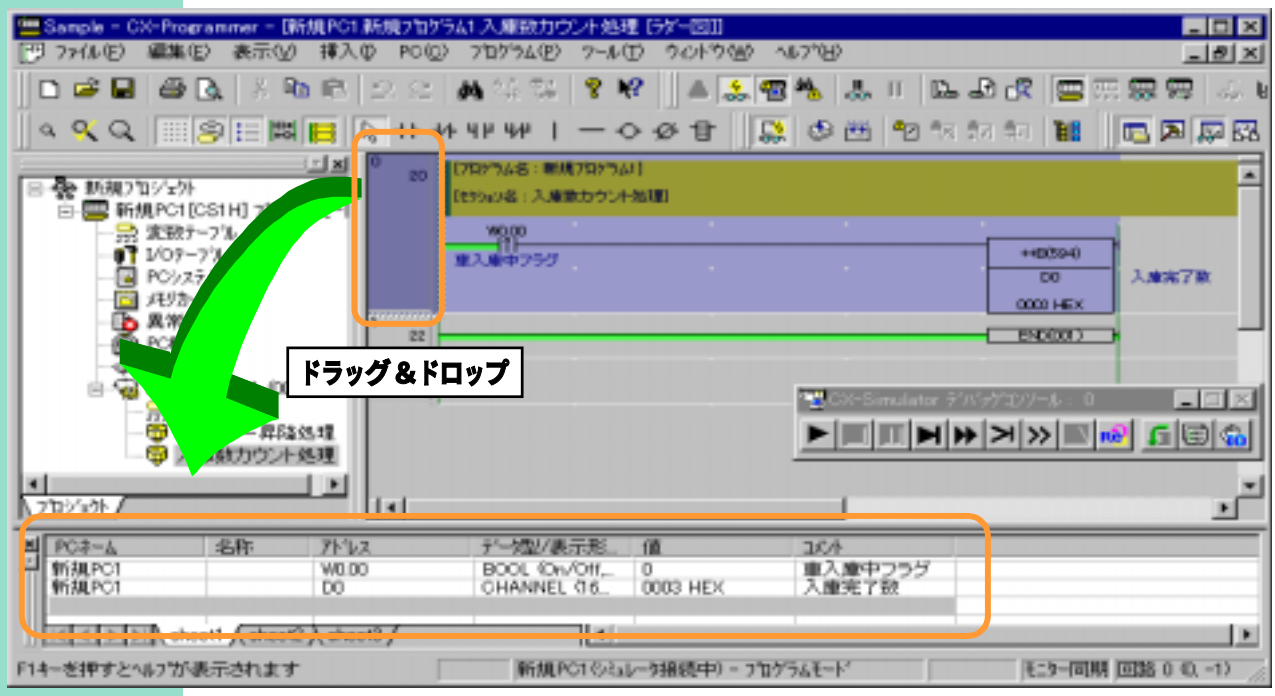
ウォッチウインドウの表示は、Alt を押しながら 3 を押します。

CX-Programmer のセット／リセット機能により、車の到着、通過の動作を行います。
ここでは、ウォッチウインドウ（I/O モニタ機能）の操作で行います。
あらかじめウォッチウインドウを表示します。

事前に仮想 PLC を実行させておきます。

ウォッチウインドウへのアドレス登録

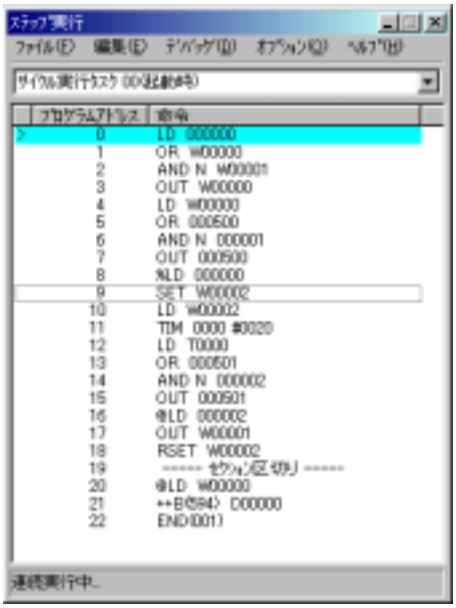
このサンプルプログラムでは、車入庫中フラグ (W0.00) の立上がりにより入庫数カウント (D0) に入庫数をカウントしていますので、ウォッチウインドウにこれら2つのアドレスを登録します。
アドレスの登録は、ラダーウインドウからのドラッグ＆ドロップの操作で行うことができます。



モニタしたいアドレスのウォッチウインドウへの登録は、ラダーウインドウからのドラッグ＆ドロップ操作で簡単に行うことができます。登録は、回路単位だけでなく、個々の接点・コイル、応用命令だけでも同様の操作で登録することができます。
また登録したアドレスに対して、セット／リセットや現在値変更の操作も簡単に行うことができます。

事前にステップ実行画面を表示しておきます。
(現在、「連続実行中」と表示されています。)

デバッグコンソール内のアイコンをクリックします。

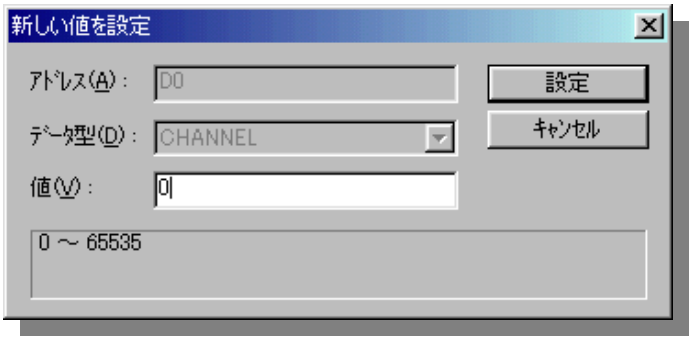


ウォッチウインドウに登録されているＤ０をダブルクリックします。
（または、Ｄ０を選択後、リターンキーを押します。）

まず、入庫完了数（Ｄ０）の値を“０”（初期化）します。

PCネーム	名称	アドレス	タイプ	値	コメント
新規PC1		W0.00	BOOL	0	車入庫中フラグ
新規PC1		D0	CHANNEL	0005 HEX	入庫完了数

値に“０”を入力し、リターンキーを押します。



Ｄ０の値が”0000 HEX“と変更されます。

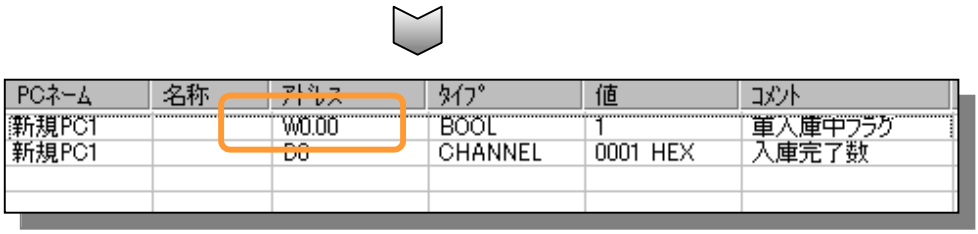
PCネーム	名称	アドレス	タイプ	値	コメント
新規PC1		W0.00	BOOL	0	車入庫中フラグ
新規PC1		D0	CHANNEL	0000 HEX	入庫完了数

I/O ブレークによる一時停止（車到着／車通過の擬似入力）

ウォッチウインドウに登録されている W0.00 をダブルクリックします。

車到着動作を行うために、W0.00 をセットします。

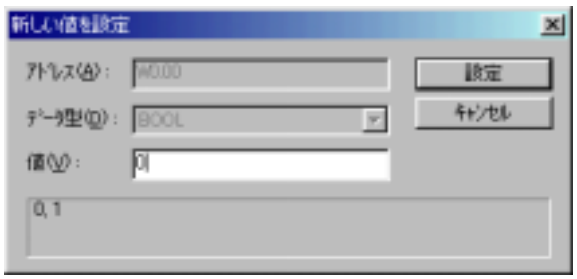
値に“1”（セット）を入力し、リターンキーを押します。



入庫完了数（D0）が“0001 HEX”に加算されます。

ウォッチウインドウに登録されている W0.00 をダブルクリックします。

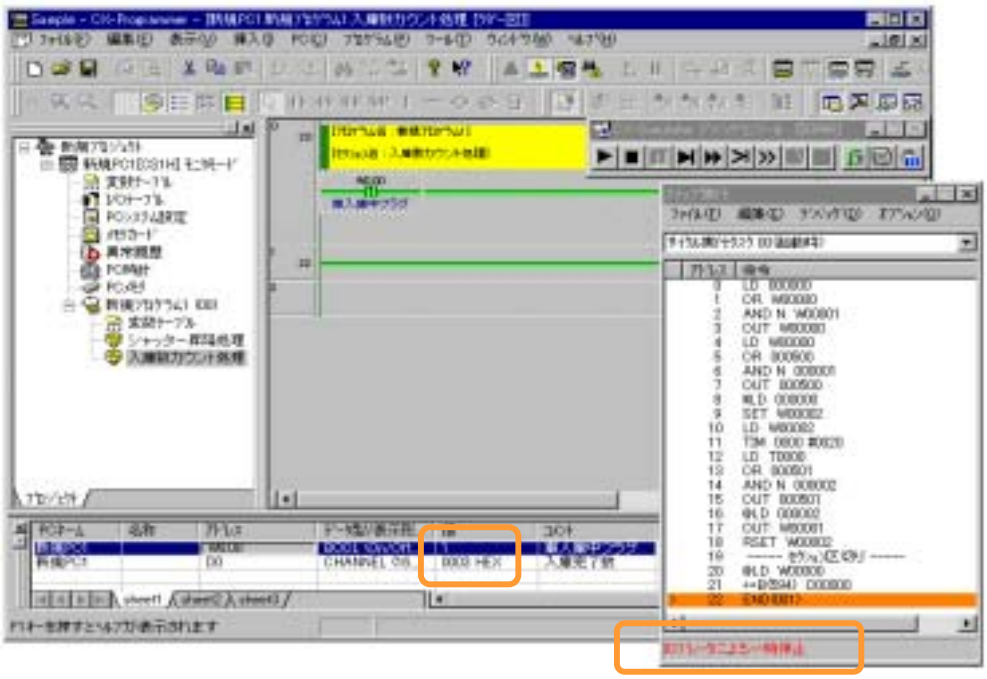
車通過動作を行うために、W0.00 をリセットします。



値に“0”（リセット）を入力し、リターンキーを押します。

D0 の値が“0003”になるまで繰り返します。

入庫完了数（D0）が“0003”になると、ステップ実行画面の下部の表示が、「I/O ブレークによる一時停止」に変化し、プログラム実行が停止します。

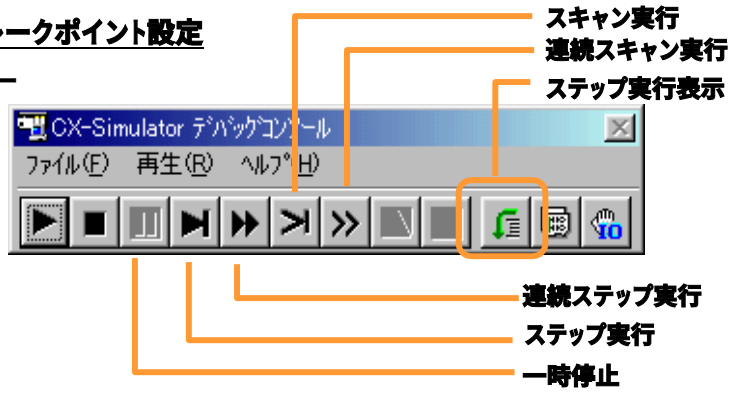


4－4　その他のデバッグ機能の紹介

プログラムのステップ単位、スキャン単位での実行、ブレークポイント設定

たとえば、ステップ実行を使うと、サブルーチン処理をコールするシーケンス検証や、FOR～NEXT 処理中の値を確認することが可能です。(図1)

また、ブレークポイントを設定しスキャン実行すると、ブレークポイントまでの実行を行ない一時停止させることが可能です。これにより任意の行から実行を再開させることが可能となります。(図2)



ステップ実行　：プログラムを1命令ずつ実行します。
連続ステップ実行　：自動的に1命令ずつ実行を進めていきます。
一時停止　：連続ステップ実行を一時停止させます。
スキャン実行　：1スキャン実行して一時停止します。
連続スキャン実行　：1スキャン実行の動作を繰り返します。

図1　ステップ実行によるデバッグ例

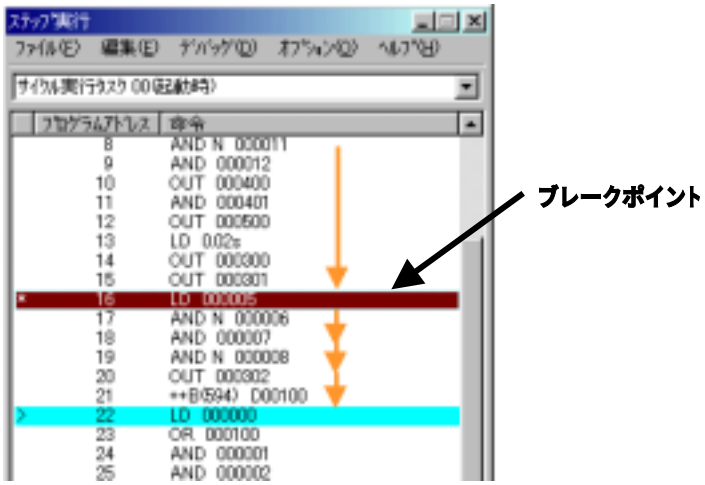


図2　ブレークポイントによるデバッグ例

第5章
CX-Simulator メニューからの起動

CX-Simulator メニューからの CX-Simulator の起動、パソコン内への仮想 PLC 作成の設定方法および CX-Simulator の終了方法について説明します。

CX-Simulator メニューから仮想 PLC を作成することにより、以下の機能を使用することができます。

操作の詳細につきましては、CX-Simulator の取扱説明書を参照してください。

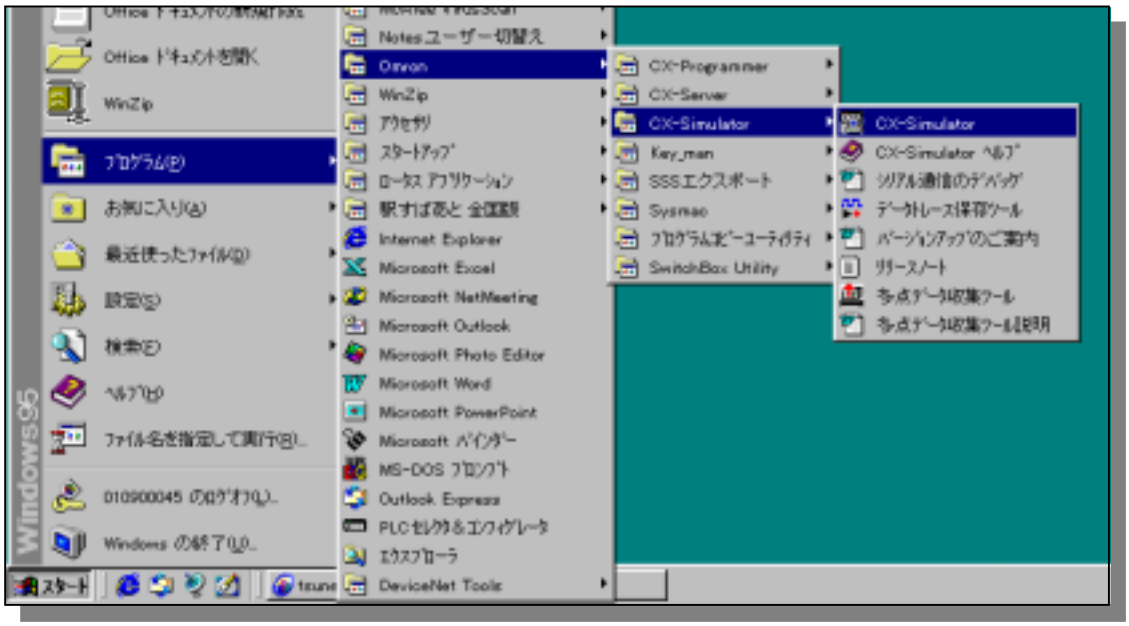
- PT などと接続するためのシリアル通信
- ネットワーク通信命令によるネットワーク通信
- I/O ユニットを登録しての I/O リフレッシュ時間の計測
- メッセージ命令やネットワーク通信命令のパソコン画面への表示
- 仮想 PLC で送受信された FINS コマンドの通信ログの記録

[スタート]
↓
[プログラム]
↓
[Omron]
↓
[CX-Simulator]
↓
[CX-Simulator]

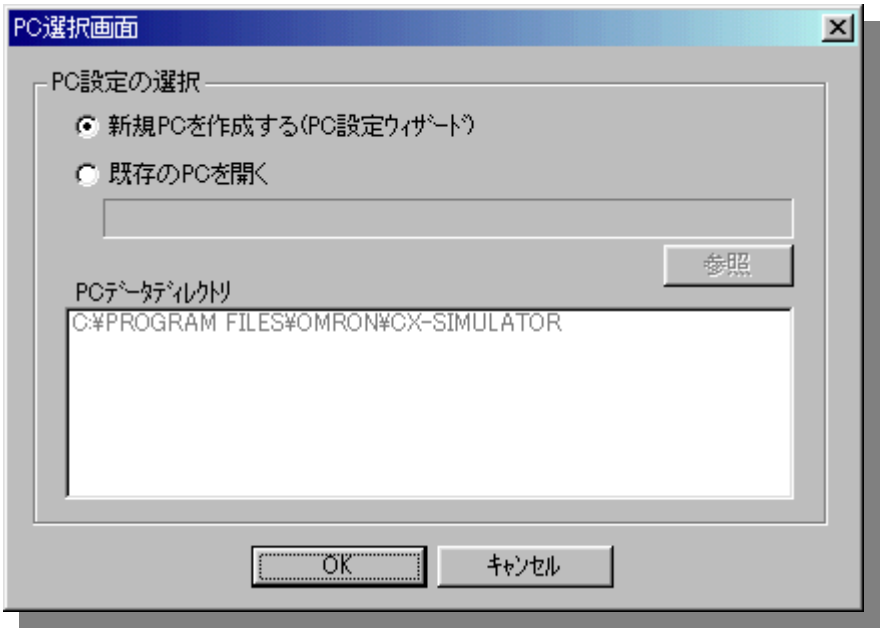
または、CX-Simulator
ｱｲｺﾝをﾀﾞﾌﾞﾙｸﾘｯｸします。

5－1　CX-Simulator の起動

Windows より、CX-Simulator を起動します。



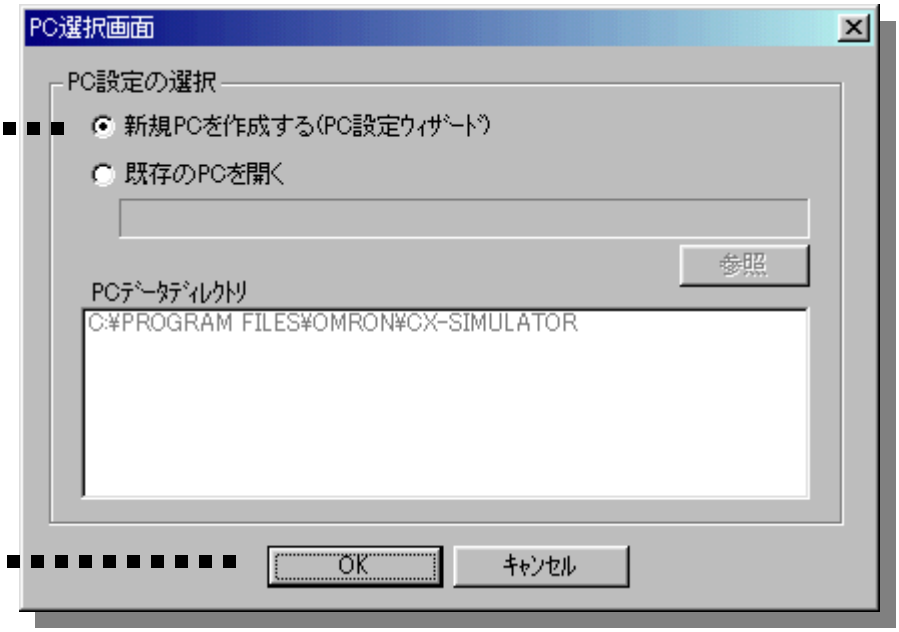
CX-Simulator の PC 選択画面とデバッグコンソールの2つの画面が表示されます。



5－2　仮想 PLC の作成

CX-Simulator 起動後に表示される PC 選択画面の内容に従い、仮想 PLC の作成を行います。

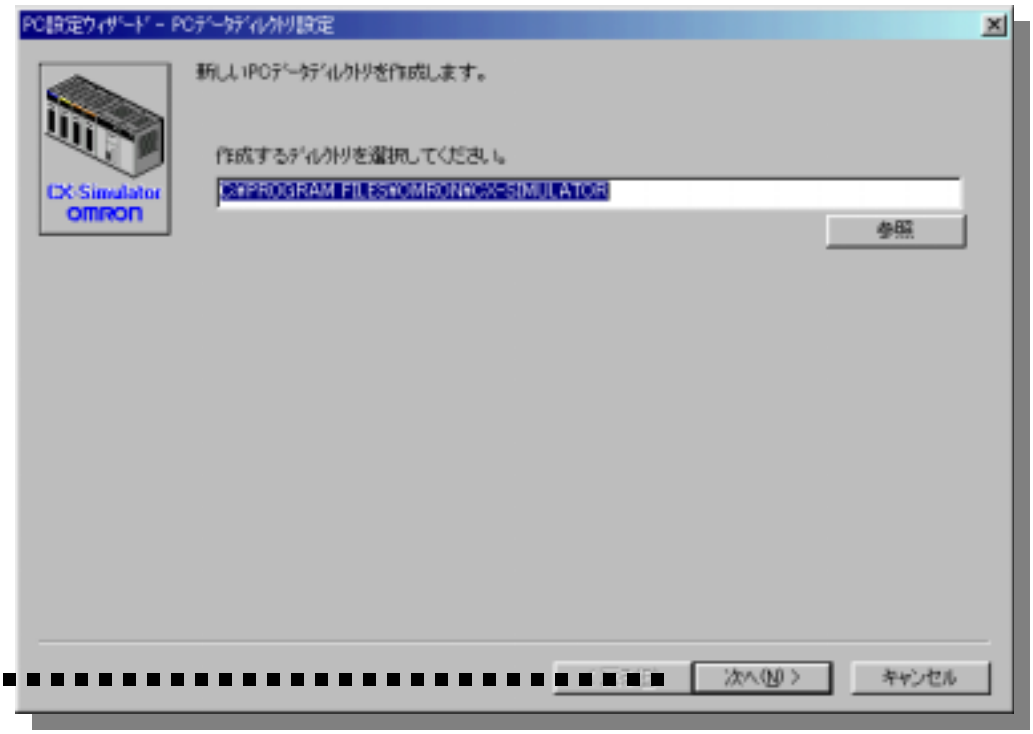
「新規PCを作成する」をチェックします。



OK

データディレクトリの指定

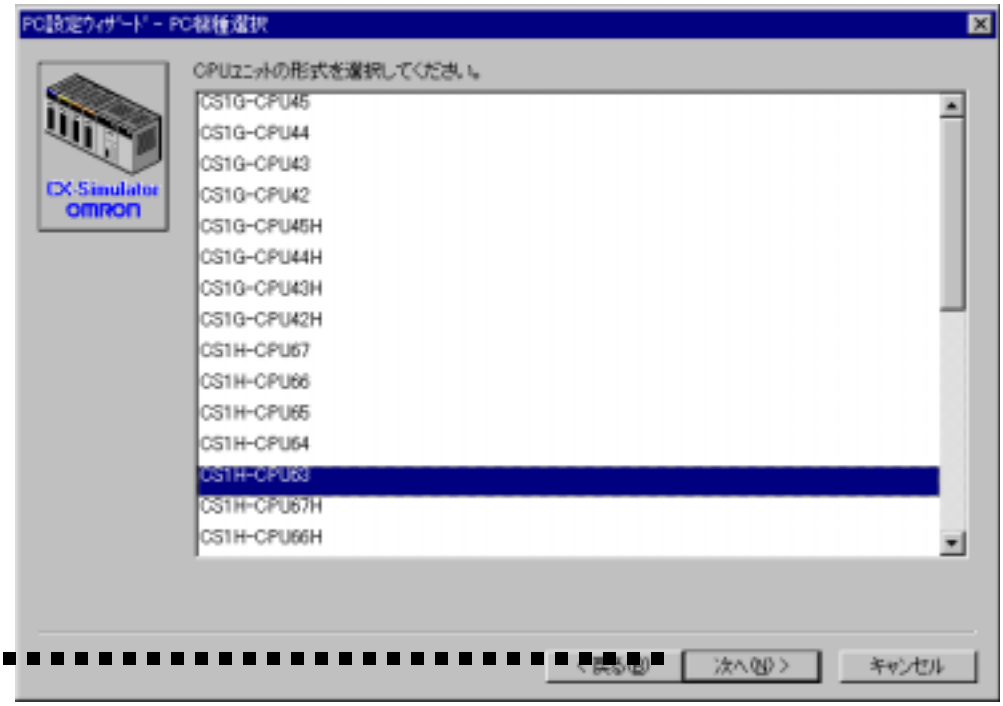
データを保存するディレクトリを設定します。ここではデフォルトのディレクトリとします。



次へ(N) >

PC機種選択

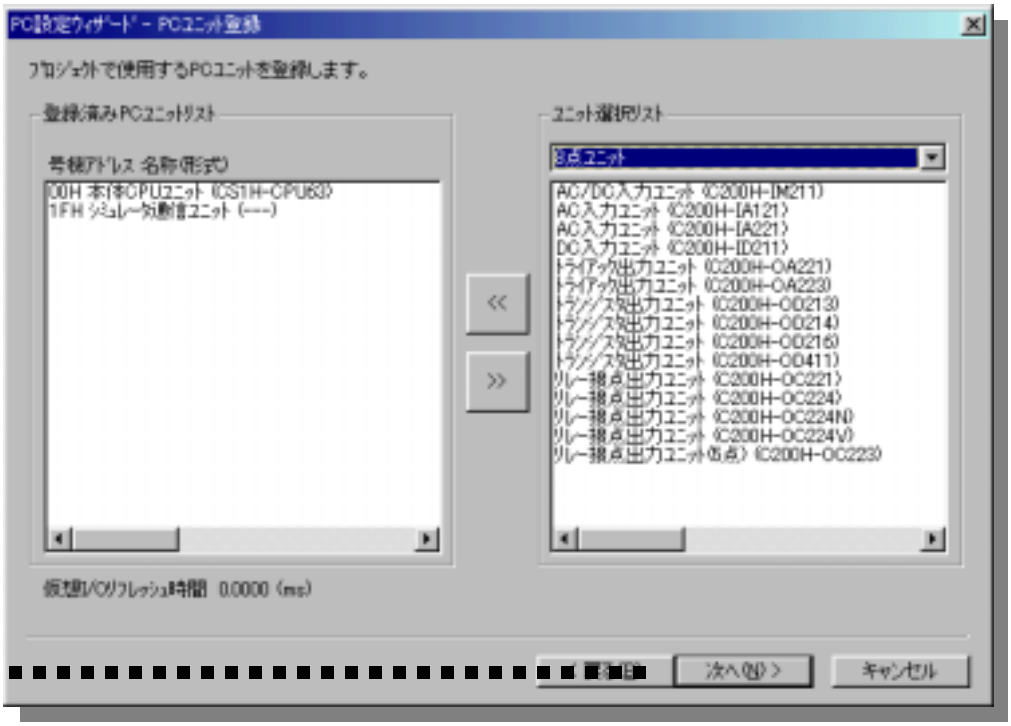
CPU ユニットの形式を指定します。ここでは、CS1H-CPU63 を指定します。



次へ(N) >

PCユニット登録

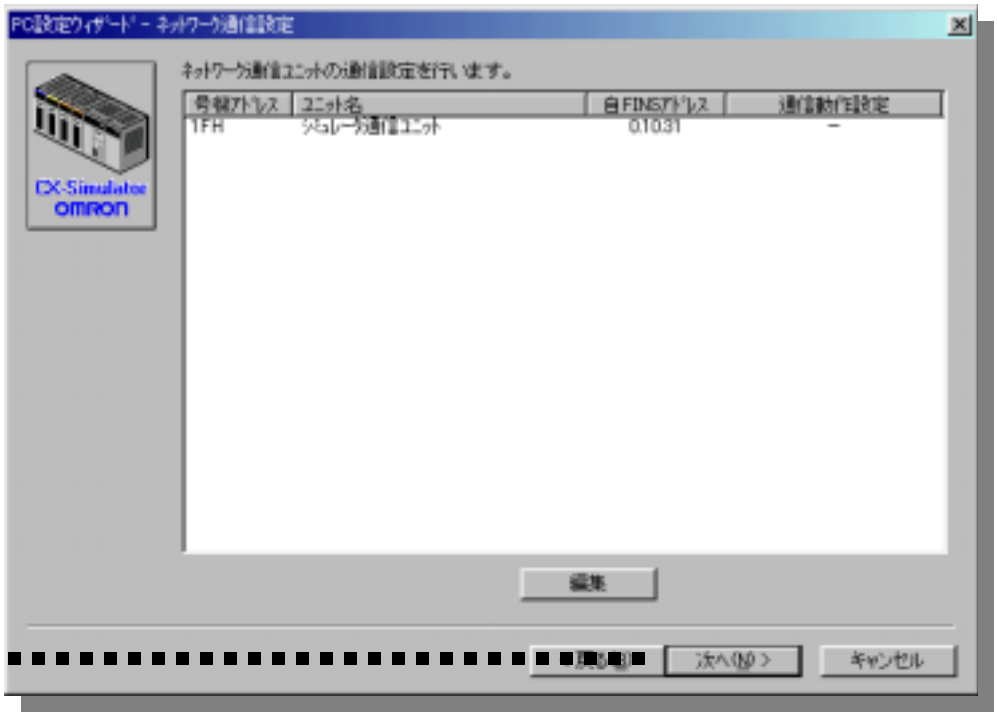
I/O ユニット、高機能ユニットを指定することができます。実際に使用するユニットを登録しておくと、サイクルタイムの事前検証の精度が上がります。ここでは、何もユニットを登録しないことにします。



次へ(N) >

ネットワーク通信設定

仮想 PLC に接続するための通信設定です。通常は変更しません。



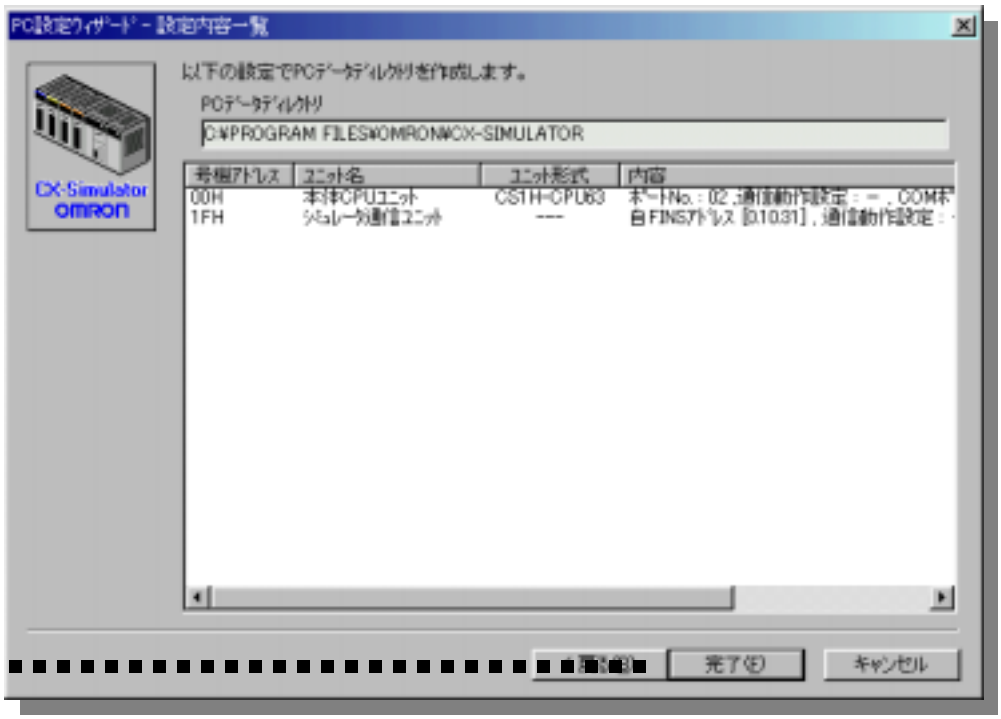
次へ(N) >

シリアル通信設定

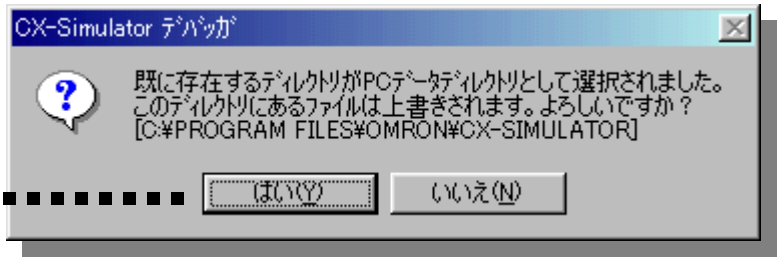
パソコンのシリアルポートを CPU ユニット内蔵のシリアルポートに見たてて、PT（プログラマブルターミナル）のような外部機器と接続することができます。ここでは、何も設定しないことにします。

次へ(N) >

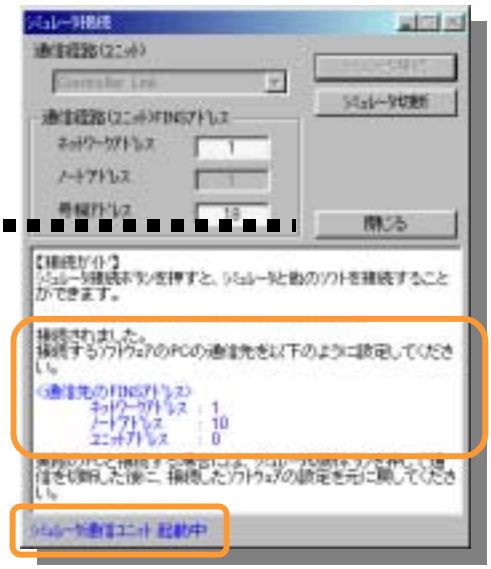
設定内容一覧


PC設定ウィザード - 設定内容一覧
以下の設定でPCデータディレクトリを作成します。
PCデータディレクトリ
C:¥PROGRAM FILES¥OMRON¥CX-SIMULATOR
号機アドレス
ユニット名
ユニット形式
内容
00H
本体CPUユニット
CS1H-CPU63
1FH
シミュレータ通信ユニット

CX-Simulator
OMRON
完了(F)
キャンセル


CX-Simulator デバッガ
既に存在するディレクトリがPCデータディレクトリとして選択されました。
このディレクトリにあるファイルは上書きされます。よろしいですか？
[C:¥PROGRAM FILES¥OMRON¥CX-SIMULATOR]
はい(Y)
いいえ(N)

シミュレータ接続
通信経路(ユニット)
Controller Link
シミュレータ接続
通信経路(ユニット)FINSアドレス
ネットワークアドレス
1
ノードアドレス
1
号機アドレス
18
閉じる
【接続ガイド】
シミュレータ接続ボタンを押すと、シミュレータと他のソフトを接続することができます。
シミュレータ通信ユニット 停止中



閉じる

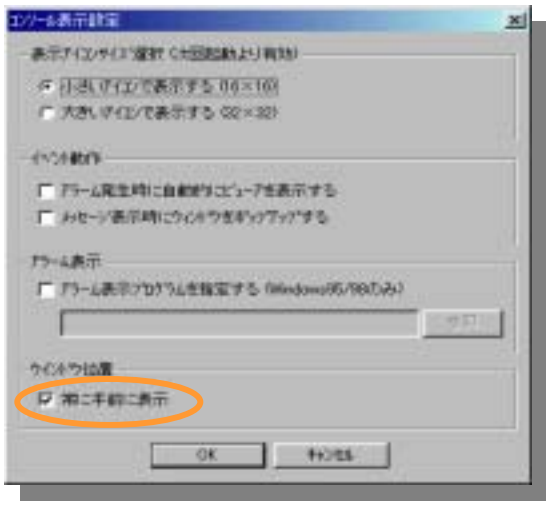
仮想 PLC が作成されたことを確認後、ウィンドウを非表示にすることができます。

これで仮想ＰＬＣの作成は完了です。
この状態で、パソコン内に仮想的な CS1H-CPU63 が存在しています。
仮想 PLC が作成されると、デバッグコンソール（上記右）の「NETWORK」が点滅し始めます。

仮想 PLC は、ネットワークアドレス：1、ノードアドレス：10として、パソコン内に作成されています。CX-Programmer Ver.3.0 以降を使用することにより、プロジェクトの PLC 機種を、この場合 CS1H-CPU63 に合わせておけば、この方法で作成した仮想 PLC に対しても、面倒な通信設定を合わせることなく、「シミュレータオンライン接続」機能が使用できます。

CX-Simulator コンソール表示設定

各種画面を設定することができます。「常に手前に表示」にチェックしておくと便利です。



デバッグコンソール画面の[ファイル]｜[システム終了]を選択します。

5－3　CX-Simulator の保存と終了

設定した仮想 PLC のデータの保存と終了方法を説明します。